

# QUANTUM TURING MACHINES

## COMPUTATIONS AND MEASUREMENTS

STEFANO GUERRINI\*, SIMONE MARTINI<sup>§</sup>, AND ANDREA MASINI<sup>¶</sup>

**ABSTRACT.** We propose a new formulation of Quantum Turing Machines, as an extension of those proposed by Bernstein and Vazirani. For this new class of Quantum Turing Machines, both finite and infinite computations are meaningful—an infinite computation does not correspond trivially to a divergent function. Moreover, we propose a natural observation protocol for the new QTMs, that does not modify the probability of the possible outcomes of the machines. Finally, we use QTMs to define a class of quantum computable functions—any such a function is a mapping from a general quantum state to a distribution of probability of natural numbers.

### 1. INTRODUCTION

Quantum computability still needs a general theory of functions, akin to the one developed at the mid of the twentieth century for classical machines, and more adherent to Feynman’s original motivations of simulating physics by computers [11].

Indeed, so far quantum computing has been studied mainly as a new, efficient paradigm for classical, discrete functions. In this way, many successes have been achieved in computational complexity (e.g., the definition of quantum complexity classes and their relations with classical ones) and algorithm design (e.g., Shor factorisation algorithm [28, 29]).

At the same time, it has been left unanswered the question of how (quantum) physics might be naturally simulated, and not just encoded, by a computing machine. In this direction, the first step is to look at quantum computation devices as machines computing more general functions than the traditional discrete functions over natural numbers (which is enough, in the classical setting, since any other discrete domain may be classically encoded into  $\mathbb{N}$ ). It is clear, however, that classical computations should remain a particular case of the quantum one, and therefore one has to cope

---

*Date:* March 23, 2017.

\* Partially supported by the Project ELICA (ref. ANR-14-CE25-0005), of the ANR program “Fondements du numérique (DS0705) 2014”.

<sup>§</sup> Partially supported by the Italian “National Group for Algebraic and Geometric Structures, and their Applications” (GNSAGA-INDAM).

<sup>¶</sup> Partially written while at LIPN, Institut Galilée, Université Paris 13, Sorbonne Paris Cité as visiting researcher.

with the unavoidable constraints established by classical computability—undecidability of program termination, first, and non-recursive enumerability of total functions, then. As a result, in a general theory of quantum computable functions one is forced to a framework where also non-total functions are present, as a results of non-terminating, or infinite, computations.

A natural starting point are Quantum Turing Machines. We are of course aware that other computational models might be considered, but we are also convinced that, in the development of a quantum computability theory, we cannot avoid QTMs. For instance, we cannot restrict to quantum circuits—which on the other hand are the simplest setting for quantum computational complexity—since circuits can only represent classes of terminating computations. Moreover, in defining families of circuits one must ensure enumerability of such families, or equivalently, that they may be computed by another device—usually, a classical Turing Machine, thus begging the question.

We therefore start with an analysis of QTMs, and in particular of the QTMs of Bernstein and Vazirani [3]. However, the standard presentation of QTMs as devices computing discrete classical functions does not fit our goal. Since unitarity of evolution forbids that a QTM may halt in a given state, several constraints are imposed to guarantee that in a computation starting on a single “natural number”, after a finite number of steps a single result may be read, thus mimicking—in a sense—terminating classical discrete computations. It is our assumption, instead, that the natural input of a quantum computation should be a generic element of the Hilbert space  $\ell^2(B)$  (for a discrete set  $B$ ;  $\mathbb{N}$  for instance), thus, in general, an infinite denumerable sum of simple configurations. Moreover, we want to allow meaningful infinite computations, where the global output is obtained only as a limit.

One of the technical contributions of this paper is a definition of QTMs in which, when a computation of the machine enters into a “final state”, its evolution continues remaining in that final state, without changing the output written on the tape. This kind of evolution is obtained by enriching the machines by means of a suitable counter which plays no role during the standard evolution of the machine, but starts to be increased when the computation enters into a final state. Indeed, as already remarked, unitary evolution of a QTM forces it to keep modifying its configuration—there cannot be a halting state. Therefore, once a final state is reached, we assume that the only possible “final evolution” of the machine is to increment by 1 the counter, leaving all the rest of the configuration (the internal state, the position of the head, and the content of the tape) unchanged.

Such an approach might remind the so-called “ancilla” proposed in the literature. However, it is known that if the ancilla is introduced in a naive way, no room is left for any interesting evolution of the QTM—see, for instance [16], whose approach leads the authors to argue that the constraint

that the output, modulo the ancilla, does not change leaves as only possible machines with unitary evolution those which never enter into a final state. Our definition of the evolution of QTMs shows instead that we can get a unitary evolution in which, once achieved, an output information is definitely preserved, allowing a computing schema in which the output is constructed incrementally. The key tool for this result is the separation of the space of the states of the final evolutions—those performed by branches of computation which have already produced an output (since they reached a final state), and which are characterised by a non zero counter—from the space of the states of the main evolution of the machine—the ones in which the input is transformed into the output, and for which the counter is set to zero. We stress that the above final behaviour cannot be restricted to a unique final state, but it must be extended to a set of generalised final states—that we name target states in the paper—that behave as sinks from which the machine cannot get out, and in which the computation continues in a fixed way that accords with the unitarity restriction, without modifying the content of the tape. In a sense, these additional target states play the same role of error states in classical deterministic TMs, allowing to complete the behaviour of a machine when it is not completely specified.

Once resolved the question of the final evolution of the machine, it remains open the question of the evolution before the initial state. Indeed, by inverting the unitary operator describing the evolution of a QTM, (possible) configurations of the machines must be defined before the initial state. To solve this problem, Bernstein and Vazirani back-connect the final states of their machines to the initial state. According to our previous discussion, such an approach is incompatible with our goal of indefinitely preserving the ability to read an output after it has been produced. Moreover, as we shall discuss in more details later, such an approach can be reasonably taken into account only in the case of the so-called well-behaved QTMs of Bernstein and Vazirani, in which all the superposed configurations of the machine enter the final state at the same time. To tackle (pre-)initial evolution, we introduce a notion of source states, dual to the target states, and among which there is an initial state in which we assume to set the machine to start a computation. As for the case of target states, this general notion of source state fixes at the same time the problem of the reverse evolution from the initial state, and the possibility to complete a partially specified machine in which, in order to satisfy the local conditions characterising the unitary condition, some additional transitions must be added to those describing the desired function. If needed, such missing transitions can be taken from source states that, being not reachable by any other state, would not interfere with the desired behaviour that we want to observe when starting from a valid configuration.

By keeping distinct the initial and the final evolution of the machine from the main one, we preserve one of the key properties of Bernstein and Vazirani's approach: QTMs can be characterised by a set of local conditions

on their transition function (see Theorem 12). Such conditions are the same of Bernstein and Vazirani. Thus, any Bernstein and Vazirani QTM  $M$  can be immediately extended into one of our QTMs whose main transition function corresponds to that of  $M$ .

In conclusion, we show that it is possible to define QTMs such that: (i) they take as input a general quantum state—that is, a denumerable superposition of simple configurations representing natural numbers; and (ii) they may have meaningful infinite behaviours, whose “output” is obtained as a limit of finite portions of the computation. We stress that we are not interested in QTMs as devices computing functions over natural numbers. Instead, we see a QTM as defining a function from a general quantum state to a distribution of probability of natural numbers. We believe this could be the starting point of a general theory of quantum computable functions.

**1.1. Content of the paper.** Despite the availability of a large corpus of results<sup>1</sup>, quantum computability still lacks a general treatment akin to classical computability theory. Taking as a reference model (quantum) Turing machines, one of the main obstacles is that while it is obvious how to understand a classical Turing machine (TM) as a device computing a numerical function, the same is not so for a quantum Turing machine (QTM).

In a naïve but suggestive way, a QTM may be described as a classical TM which, at any point of its computation, evolves into several different classical configurations, each characterised by a complex amplitude. Such different configurations should be imagined as simultaneously present, “in superposition”—a simultaneity formally expressed as a weighted sum  $\sum d_i C_i$  of classical configurations  $C_i$ , with complex coefficients  $d_i$ . Even when starting from a classical configuration, in a QTM, there is not a single result, but a superposition from which we can read off several numerical “results” with certain probabilities<sup>2</sup>. Moreover, QTMs never have genuinely finite computations and one should therefore find a way to define when and how a result can be read off.

We propose a notion of “function computable by a quantum Turing machine,” as a mapping between superpositions of initial classical configurations<sup>3</sup> to probability distributions of natural numbers, which are obtained (in general) as a limit of an infinite QTM computation.

Before reaching this point, however, we must go back to the basics, and look to the very notion of QTM. Because, if it is true that configurations of a QTM are superpositions  $\sum d_i C_i$  of classical configurations, quantum physics

<sup>1</sup>See, in the large literature, [10, 18, 20, 23, 30] for fundamentals results, [3] for the foundations of quantum complexity, or [1, 15, 6, 7, 5, 8, 17, 26, 27, 31] for more language oriented papers.

<sup>2</sup>We cannot observe the entirety of a superposition without destroying it. But if we insist on observing it, then we will obtain one of the classical configurations  $C_i$ , with probability  $|d_i|^2$ .

<sup>3</sup>More precisely, the domain of our functions will be the Hilbert space  $\ell_1^2(\mathbb{N})$  of square summable, denumerable sequences of complex numbers, with unitary norm.

principles impose severe constraints on the possible evolutions of such machines. First, in any superposition  $\sum d_i C_i$ , we must have  $\sum |d_i|^2 = 1$ . Second, there cannot be any finite computation—we may of course name a state as “final” and imagine that we read the result of the computation (whatever this means) when the QTM enters such a state, but we must cope with the fact that the computation will go on afterwards. Moreover, since any computation of a QTM must be reversible [2], in the sense that the operator describing the evolution of the QTM must be invertible, we cannot neither force the machine to loop on its final configuration. On the other hand, because of reversibility, even the initial configuration must have a predecessor. Summing up, an immediate consequence of all the above considerations is that every state must have at least one incoming and one outgoing transition and that such transitions must accord to several constraints forced by quantum physics principles. In particular, transitions must enter the initial state—since a priori it might be reached as the evolution from a preceding configuration — and exit the final state—allowing the machine configuration to evolve even after it has reached the final result.

The reversibility physical constraints are technically expressed by the requirement that the evolution operator of a QTM be unitary. If we now want to use a QTM to compute some result, we are still faced with the problem of when (and how, but for the moment let’s postpone this) reading such a final result, given that the computation evolves also from the final state, and that, without further constraints, it might of course evolve in many, different possible ways. Bernstein and Vazirani in their seminal paper [3] (from now on we shall refer to this paper as “B&V”) first define (non unitary) QTMs; select then the “well-formed” QTMs as the unitary-operator ones; and define finally “well-behaved” QTMs as those which produce a superposition in which all classical configurations are simultaneously (and for the first time) in the final state. What happens after this moment, it is not the concern of B&V.

Our goal is to relax the requirement of simultaneous “termination”, allowing meaningful computations to reach superpositions in which some classical configurations are final (and give a “result”), and some are not. Those which are not final, should be allowed to continue the computation, possibly reaching a final state later. The “final result” will then be defined as a limit of this process. In order to assure that at every step of the computation the superposition of the final configurations is a valid approximation of the limit result, we must guarantee that once a final state is entered, the “result” that we read off is not altered by the further (necessary, by unitarity) evolution. To obtain this, we restrict the transition function out of a final state, without violating unitarity. We obtain this goal by using an integer counter:

- Once a final state is entered, the counter starts counting the number of steps during which the computation has been looping into the final state; that is, when in a final state, at each step, the machine just

increases the counter by 1, leaving unchanged the state, the tape, and the position of the head.

- Dually, in an initial state, the counter gives the number of steps before the machine might actually start its main computation. That is, when a machine is in initial state with a counter greater than 0, it rollbacks to another initial configuration which differ for the value of the counter only, since it has been decreased by 1.
- Finally during the normal or main evolution (i.e. the quantum evolution as defined by B&V) the counter does not play any role and its value is zero.

In the paper we will apply this approach to a generalisation of initial/final states, called here *source* and *target* states.

After the definition of QTMs and of the corresponding functions, we will discuss their expressive power, comparing them to the QTMs studied in the literature. The QTMs of B&V form a robust class, but meaningful computations are defined only for classical inputs (a single natural number with amplitude 1). Moreover, their QTMs “terminate” synchronously—either all paths in superpositions enter a final state at the same time, or all of them diverge. As a consequence, there is no chance to study—and give meaning—to infinite computations. More important, the class of “sensible” QTMs (in B&V’s terminology: the well-formed, well-behaved, normal form QTMs) is not recursively enumerable, since the constraint of “simultaneous termination” is undecidable.

In Deutsch’s original proposal [10], any quantum TM has an explicit termination bit which the machine sets when entering a final configuration. While it is guaranteed that final probabilities are preserved, the observation protocol requires that one checks termination at every step, since the machine may well leave the final state (and change the tape). Deutsch’s machines could in principle be used to define meaningful infinite computations, but we know of no such an attempt.

In our analysis: (i) there is no termination bit: a quantum configuration is a genuine superposition of classical configurations; (ii) any computation path (that is, any element of the superposition) evolves independently from the others: any path may terminate at its own time, or may diverge; (iii) infinite computations are meaningful; (iv) we may observe the result of the computation in a way akin to Deutsch’s one, but with the guarantee that once a final state is entered, the machine will not change the corresponding “result” during a subsequent computation; (v) the class of QTMs is recursively enumerable, thus opening the door to a quantum computability theory which may follow some of the classical developments.

## 2. QUANTUM TURING MACHINES

In this section we define quantum Turing machines. We assume the reader be familiar with classical Turing machines (in case, see [9]).

As for classical Turing Machine (TM), a Quantum Turing Machine (QTM) has a tape (a sequence of cells) containing symbols (one in each cell) from a finite tape alphabet  $\Sigma$ , which includes at least the symbols 1 and  $\square$ : 1 is used to code natural numbers in unary notation, while  $\square$  is the blank symbol. We shall consider computations starting from tapes containing a sequence of  $n + 1$  symbols 1 (the encoding of the natural number  $n$ ); thus, in the following, for any  $n \in \mathbb{N}$ , we shall use  $\underline{n}$  to denote the string  $1^{n+1}$ . By the greek letters  $\alpha, \beta$ , eventually indexed, we shall instead denote strings in  $\Sigma^*$ , and by  $\alpha\beta$  we shall denote the concatenation of  $\alpha$  and  $\beta$ . Finally, we shall use  $\lambda$  to denote the empty string.

**2.1. Plain configurations.** The basic elements to describe the configuration of a QTM are the finite sequences of symbols on the tape (as usual, we assume that only a finite portion of the tape contains non-blank symbols), the current internal state of the machine, and the current position of the head reading a symbol in a cell of the tape. More precisely, a plain configuration of a given QTM  $M$  is a triple  $\langle \alpha, q, \beta \rangle \in \Sigma^* \times Q \times \Sigma^*$ , s.t.:

- (1)  $q \in Q$  is the current state, where  $Q$  is the finite set of the internal states of  $M$ ;
- (2)  $\beta \in \Sigma^+$  is the right content of the tape, where  $\Sigma$  is the tape alphabet of the machine  $M$ . The first symbol of  $\beta$  is the one in the current cell of the tape, that is, the one read by the head of  $M$ . In detail,  $\beta = u\beta'$ , where the current symbol  $u$  is the content of the current cell and  $\beta'$  is the longest string on the tape ending with a symbol different from  $\square$  and whose first symbol (if any) is written in the cell immediately to the right of the current cell; by convention, when the current cell and all the right content of the tape is empty, we shall also write  $\langle \alpha, q, \lambda \rangle$  instead of  $\langle \alpha, q, \square \rangle$ ;
- (3)  $\alpha$  is the left content of the tape. That is, it is either the empty string  $\lambda$ , or it is the longest string on the tape starting with a symbol different from  $\square$ , and whose last symbol is written in the cell immediately to the left of the current cell.

According to this definition, in a configuration  $\langle \alpha, q, \beta \rangle$  the string  $\alpha$  does not start with a  $\square$ , and  $\beta$  does not end with a  $\square$ . However, since it will be useful to manipulate configurations which are extended with blank cells to the right (of the right content) or to the left (of the left content), we equate configurations up to the three equivalence relations induced by the following equations

$$\begin{aligned} \alpha &\simeq_l \square\alpha & \beta &\simeq_r \beta\square \\ \langle \alpha, q, \beta \rangle &\simeq \langle \alpha', q, \beta' \rangle & \text{when } \alpha &\simeq_l \alpha' \text{ and } \beta \simeq_r \beta' \end{aligned}$$

**2.2. Hilbert space of configurations.** A quantum configuration of a QTM is not a simple configuration as the ones of a TM, but a weighted superposition of configurations: a vector of the Hilbert space  $\ell^2(\mathfrak{C})$ , where  $\mathfrak{C}$  is a suitable set of simple configurations as the plain ones defined above.

We recall that, for any denumerable set  $\mathcal{B}$ ,  $\ell^2(\mathcal{B})$  is the Hilbert space of square summable  $\mathcal{B}$ -indexed sequences of complex numbers

$$\left\{ \phi : \mathcal{B} \rightarrow \mathbb{C} \mid \sum_{C \in \mathcal{B}} |\phi(C)|^2 < \infty \right\}$$

equipped with an inner product  $\langle \cdot \mid \cdot \rangle$  and the euclidean norm  $\|\phi\| = \sqrt{\langle \phi \mid \phi \rangle}$ , and that  $\ell^2_1(\mathcal{B})$  denote the set of vectors  $\{\phi \mid \phi \in \ell^2(\mathcal{B}) \text{ \& } \|\phi\| = 1\}$ .

By using Dirac notation, we shall write  $|\phi\rangle$  to denote the vector of  $\ell^2(\mathcal{B})$  corresponding to the function  $\phi : \mathcal{B} \rightarrow \mathbb{C}$ . Moreover, for every  $C \in \mathcal{B}$ , we shall write  $|C\rangle$  to denote the vector corresponding to  $C$ , that is, the function equal to 1 on  $C$ , and equal to 0 on the other elements of  $\mathcal{B}$ . Finally, we remark that, any vector  $|\phi\rangle$  of  $\ell^2(\mathcal{B})$  can be written as  $\sum_{i \in I} d_i |C_i\rangle$ , for some denumerable set of indexes  $I$  s.t.  $\{C_i \mid i \in I\} \subseteq \mathcal{B}$  and  $d_i \in \mathbb{C}$ , for every  $i \in I$ . For more details on the basic notions of Hilbert spaces, see Appendix A.

**2.3. Transitions of a QTM.** Given a quantum configuration  $\phi$  of a QTM  $M$ , the main idea introduced by Deutsch [10] is that the machine evolves into another quantum configuration  $|\psi\rangle = U|\phi\rangle$ , where  $U$  is a unitary operator on the Hilbert space of the configurations of  $M$ . By linearity, this also means that, if  $\mathfrak{C}$  is the set of simple configurations on which we define the Hilbert space of quantum configuration  $\ell^2(\mathfrak{C})$  of  $M$ , the operator  $U$ , and then the behaviour of  $M$ , is completely determined by the value of  $U$  on the elements of  $\mathfrak{C}$ . B&V [3] refine this point by assuming that, as in a classical TM, for every  $C \in \mathfrak{C}$ , the transition from  $|C\rangle$  to a new configuration  $U|C\rangle$  depends only on the current state of  $M$  and on the current symbol of  $C$ , and that  $U|C\rangle$  is formed of configurations obtained by replacing the current symbol  $u$  of  $C$  with a new one, and by moving the tape head to the left or the right. Therefore, if we denote by  $\mathbb{D} = \{L, R\}$  the set of the possible movements of the tape (where  $L$  stands for left and  $R$  for right), and  $q$  is the current state of  $M$ , we have

$$U(|C\rangle) = \sum_{p,v,d} \delta(q,u)(p,v,d) |C_{p,v,d}\rangle$$

where  $p$  ranges over the states of  $M$ ,  $v$  ranges over its tape alphabet,  $\delta(q,u)(p,v,d) \in \mathbb{C}$ , and  $C_{p,v,d}$  is the new configuration obtained from  $C$  by changing the current state from  $q$  to  $p$ , by replacing the current symbol  $u$  with  $v$ , and by moving the tape head in the direction  $d$ .

**2.4. Initial and final configurations.** In B&V, the transition rule described above applies to every state of the machine. However, as already remarked in the introduction, this implies a severe restriction on the machines that one can actually consider as valid ones, since the problem of how to read the output forces to ask that, in a “well-behaved” QTM, all the states of the configuration in superposition be final.

The key point is that a QTM cannot stop into some final configuration  $C_f$ , since the unitarity of  $U$  requires that  $U(|C_f\rangle)$  be defined. On the other



hand, we cannot resort to the assumption that after reaching some final configuration  $C_f$ , the computation loops on it, as this would imply that  $C_f$  could not be reached from any other configuration  $D \in \mathfrak{C}$ : if  $U(|C_f\rangle) = |C_f\rangle$ , then  $|C_f\rangle = U^{-1}(|C_f\rangle)$ , and  $U(|D\rangle)(C_f) = \langle U(|D\rangle) | C_f \rangle = \langle D | U^{-1}(|C_f\rangle) \rangle = \langle D | C_f \rangle = 0$ , for  $D \neq C_f$ . Because of this, when a final configuration  $C_f$  is reached, the machine must evolve into a different configuration which preserves the output written on the tape, preserving at the same time the unitarity of the evolution operator  $U$ .

To solve the above problem, we assume that, for every final configuration  $C_f$  we have a denumerable set of indexed configurations  $\langle C_f, n \rangle$ , obtained by adding a counter  $n \in \mathbb{N}$  to  $C_f$ . The configuration  $\langle C_f, 0 \rangle$  plays the usual role of the plain configuration  $C_f$  and is the only one that can be reached from a non final configuration; for every  $n \in \mathbb{N}$ , the configuration  $\langle C_f, n+i \rangle$  is instead the only successor of  $\langle C_f, n \rangle$ , that is,  $U(|C_f, n\rangle) = |C_f, n+1\rangle$ , where  $|C_f, n\rangle$  is the base vector corresponding to  $\langle C_f, n \rangle$ . In other words, the counter  $n$  is initialised to 0 and plays no role until the state is not final; when a branch of the computation enters a final state, the counter is still at 0, but it is increased by 1 at each following step.

Final configurations are not the only configurations on which it would be useful to loop. In fact, in some cases, in order to assure that the operator  $U$  is unitary, we need to introduce additional target configurations that behave as sinks, from which the machine cannot get out (see the example in subsection 5.3, whose graph representation is given in Figure 4). We have then to consider a whole set  $\mathcal{Q}_t$  of target states, containing the final state  $q_f$ , s.t. for every plain configuration  $C_t = \langle \alpha, q_t, \beta \rangle$ , with  $q_t \in \mathcal{Q}_t$ , we have a set of indexed configurations  $\langle C_t, n \rangle$ , for  $n \in \mathbb{N}$ , s.t.  $U(|C_t, n\rangle) = |C_t, n+1\rangle$ .

Finally, let us remark that when  $U$  is unitary, its inverse  $U^{-1}$  is unitary too, and can be applied to any initial configuration. Therefore, even if we assume that a computation always starts from some initial configuration  $C_i = \langle \alpha, q_i, \beta \rangle$ , such a  $C_i$  must have a predecessor  $C_{i,1} = U^{-1}(|C_i\rangle)$ , and more generally a  $n$ -predecessor  $C_{i,n} = U^{-n}(|C_i\rangle)$ . As already done for final configurations, we associate to every  $C_i$  a set of indexed configurations  $\langle C_i, n \rangle$ , with  $n \in \mathbb{N}$ , s.t.  $U^{-1}(|C_i, n\rangle) = |C_i, n+1\rangle$ , for  $n \in \mathbb{N}$ . For  $n > 0$ , we get then  $U(|C_i, n\rangle) = |C_i, n-1\rangle$ ; while  $U(|C_i, 0\rangle)$  has the usual behaviour expected from the machine on the plain initial configuration  $C_i$ . As for final state and final configurations, it is useful to generalise the above behaviour to a set of source configurations corresponding to a set  $\mathcal{Q}_s$  of source states which contains the initial state  $q_i$ .

**2.5. Pre Quantum Turing Machines.** In order to formally define QTMs, we define first a notion of pre Quantum Turing Machine, or pQTM. As we shall see later (Definition 7), a QTM is a pQTM whose evolution operator is unitary. Since the behaviour on target states and on source states whose counter is greater than 0 is fixed, in order to completely describe a pQTM, it suffices to give a transition function  $\delta_0 : ((\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma) \rightarrow \ell_1^2((\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D})$

which, for every configuration  $C$  with current state  $q$  and current symbol  $u$ , gives the weight  $\delta_0(q, u)(p, v, d)$  of  $|C_{p,v,d}\rangle$  in the superposed configuration reached from  $|C\rangle$ , where  $C_{p,v,d}$  is obtained by replacing  $v$  for  $u$ , by moving the tape in the  $d$  direction, and by changing the current state to  $p$ .

**Definition 1** (Pre Quantum Turing Machine). *Given a finite set of states  $\mathcal{Q}$  and an alphabet  $\Sigma$ , a pre Quantum Turing Machine (pQTM) is a tuple*

$$M = \langle \Sigma, \mathcal{Q}, \mathcal{Q}_s, \mathcal{Q}_t, \delta_0, q_i, q_f \rangle$$

where

- $\mathcal{Q}_s \subseteq \mathcal{Q}$  is the set of source states of  $M$ , and  $q_i \in \mathcal{Q}_s$  is a distinguished source state named the initial state of  $M$ ;
- $\mathcal{Q}_t \subseteq \mathcal{Q}$  is the set of target states of  $M$ , and  $q_f \in \mathcal{Q}_t$  is a distinguished target state named the final state of  $M$ ;
- $\delta_0 : ((\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma) \rightarrow \ell^2((\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D})$  is the quantum transition function of  $M$ , where  $\mathbb{D} = \{L, R\}$ .

**2.6. Configurations.** We have already said that, in order to properly deal with source and target states, we have to associate a counter to them. For the sake of uniformity, we shall add a counter to every plain configuration. Even if, its value get stuck to 0 for every non-source or non-target configuration.

*Remark 2* (The counter). The counter can be seen as an additional device (for instance, as an additional tape or as a counting register) or directly implemented by a suitable extension of the basic Turing machine (for instance, by extending the tape alphabet). None of these implementations is standard or has a direct influence in what will be presented in the following. The key issue pointed out in the above discussion is that, for every target configuration  $C$  of the QTM  $M$ , we need to have a denumerable space of configurations isomorphic to  $\{C\} \times \mathbb{N}$ , in which the evolution of  $M$  from  $C$  is confined, and s.t.  $U_M^k(|C\rangle)$  is in bijection with the  $k$ -th element  $\langle C, k \rangle$  of this space. Analogously, for source configurations, when considering the inverse time evolution operator  $U_M^{-1}$ . A more detailed discussion of the implementation of the counter is given in Appendix B.

**Definition 3** (configurations). *Let  $M = (\Sigma, \mathcal{Q}, \mathcal{Q}_s, \mathcal{Q}_t, \delta_0, q_0, q_f)$  be a pQTM. A configuration of  $M$  is a quadruple  $\langle \alpha, q, \beta, n \rangle \in \Sigma^* \times \mathcal{Q} \times \Sigma^* \times \mathbb{N}$ , where  $\langle \alpha, q, \beta \rangle$  is a plain configuration, and  $n$  is a counter associated to the configuration s.t.  $n = 0$ , when  $q \notin \mathcal{Q}_s \cup \mathcal{Q}_t$ . A configuration of  $M$  is a source/target configuration when the corresponding state is a source/target state, and it is a final/initial configuration when the current state is final/initial. We have the following notations:*

- $\mathfrak{C}_M$  is the set of the configurations of  $M$ .
- $\mathfrak{C}_M^s$  and  $\mathfrak{C}_M^t$  are the sets of the source and of the target configurations of  $M$ , respectively
- $\mathfrak{C}_M^{\text{init}}$  and  $\mathfrak{C}_M^{\text{fin}}$  are the sets of the initial and final configurations of  $M$ , respectively.

- $\mathfrak{C}_M^0$  is the of the configurations  $\langle \alpha, q, \beta, 0 \rangle$  of  $M$ .

In the following, the index  $M$  in  $\mathfrak{C}_M$  and in the other names indexed by the machine may drop when clear from the context.

**2.7. Quantum configurations.** The evolution of a pQTM is described by superpositions of configurations in  $\mathfrak{C}_M$ . If  $\mathcal{B} \subseteq \mathfrak{C}_M$  is a set of configurations, a superposition of configurations in  $\mathcal{B}$  is a vector of the Hilbert space  $\ell^2(\mathcal{B})$  (see, e.g., [4, 25]). Quantum configurations of a pQTM  $M$  are the elements of  $\ell_1^2(\mathfrak{C}_M)$  (namely, the unit vectors of  $\ell^2(\mathfrak{C}_M)$ ). Since there is no bound on the size of the tape in a configuration, the set  $\mathfrak{C}_M$  is infinite and the Hilbert space of the configurations  $\ell^2(\mathfrak{C}_M)$  is infinite dimensional.

**Definition 4** (quantum configurations). *Let  $M$  be a pQTM. The elements of  $\ell_1^2(\mathfrak{C}_M)$  are the  $q$ -configurations (quantum configurations) of  $M$ .*

We shall use Dirac notation (see Appendix A) for the elements  $\phi, \psi$  of  $\ell^2(\mathfrak{C}_M)$ , writing them  $|\phi\rangle, |\psi\rangle$ .

**Definition 5** (computational basis). *For any set of configurations  $\mathcal{B} \subseteq \mathfrak{C}_M$  and any  $C \in \mathcal{B}$ , let  $|C\rangle : \mathcal{B} \rightarrow \mathbb{C}$  be the function*

$$|C\rangle(D) = \begin{cases} 1 & \text{if } C = D \\ 0 & \text{if } C \neq D. \end{cases}$$

*The set  $\text{CB}(\mathcal{B})$  of all such functions is a Hilbert basis for  $\ell^2(\mathcal{B})$  (see, e.g., [20]). In particular, following the literature on quantum computing,  $\text{CB}(\mathfrak{C}_M)$  is called the computational basis of  $\ell^2(\mathfrak{C})$ . Each element of the computational basis is called base  $q$ -configuration.*

With a little abuse of language, we shall write  $|C\rangle \in |\phi\rangle$  when  $\phi(C) \neq 0$ . The span of  $\text{CB}(\mathcal{B})$ , denoted by  $\text{span}(\text{CB}(\mathcal{B}))$ , is the set of the finite linear combinations with complex coefficients of elements of  $\text{CB}(\mathcal{B})$ ;  $\text{span}(\mathcal{B})$  is a vector space, but not a Hilbert space. In order to get a Hilbert space from  $\text{span}(\text{CB}(\mathcal{B}))$  we have to complete it, and  $\ell^2(\mathcal{B})$  is indeed the unique (up to isomorphism) completion of  $\text{span}(\text{CB}(\mathcal{B}))$  (see [3]). As a consequence, any linear operator  $U$  on  $\text{span}(\text{CB}(\mathcal{B}))$  has a unique extension on  $\ell^2(\mathcal{B})$ , and, when  $U$  is unitary, its extension is unitary too.

For some basic definitions, properties and notations on Hilbert spaces with denumerable basis, see Appendix A. In particular, subsection A.1 presents a synoptic table of the so-called Dirac notation that we shall use in the paper.

**2.8. Time evolution operator and QTM.** In order to define the evolution operator of a pQTM  $M$ , it suffices to give its behaviour on the computational basis  $\text{CB}(\mathfrak{C}_M)$ . In particular, we have to distinguish three cases:

- (1)  $C \in \mathfrak{C}_M^0 \setminus \mathfrak{C}_M^t$ .

Let  $C_{p,v,d} \in \mathfrak{C}_M^0 \setminus \mathfrak{C}_M^s$  be the configuration obtained by leaving to 0 the counter, by replacing the symbol  $u$  in the current cell with the symbol  $v$ , by moving the head on the  $d$  direction, and by setting the

machine into the new state  $p$ . In detail, if  $C \simeq \langle \alpha cw, q, u\beta, 0 \rangle$  we have

$$C_{p,v,d} \simeq \begin{cases} \langle \alpha wv, p, \beta, 0 \rangle & \text{when } d = R \\ \langle \alpha, p, wv\beta, 0 \rangle & \text{when } d = L. \end{cases}$$

and we define

$$W_{0,M}(|C\rangle) = \sum_{(p,v,d) \in (\mathbb{Q} \setminus \mathbb{Q}_s) \times \Sigma \times \mathbb{D}} \delta_0(q, u)(p, v, d) |C_{p,v,d,k}\rangle$$

where  $\delta_0$  is the quantum transition function of  $M$ .

(2)  $C \in \mathfrak{C}_M^s \setminus \mathfrak{C}_M^0$ .

Let  $C_{-1} \in \mathfrak{C}_M^s$  be the source configuration obtained by decreasing by 1 the counter of  $C$ , we define

$$W_{s,M}(|C\rangle) = |C_{-1}\rangle$$

(3)  $C \in \mathfrak{C}_M^t$ .

Let  $C_{+1} \in \mathfrak{C}_M^t \setminus \mathfrak{C}_M^0$  be the target configuration obtained by increasing by 1 the counter of  $C$ , we define

$$W_{t,M}(|C\rangle) = |C_{+1}\rangle$$

We have then three linear operators

$$W_{0,M} : \text{span}(\text{CB}(\mathfrak{C}_M^0 \setminus \mathfrak{C}_M^t)) \rightarrow \text{span}(\text{CB}(\mathfrak{C}_M^0 \setminus \mathfrak{C}_M^s))$$

$$W_{s,M} : \text{span}(\text{CB}(\mathfrak{C}_M^s \setminus \mathfrak{C}_M^0)) \rightarrow \text{span}(\text{CB}(\mathfrak{C}_M^s))$$

$$W_{t,M} : \text{span}(\text{CB}(\mathfrak{C}_M^t)) \rightarrow \text{span}(\text{CB}(\mathfrak{C}_M^t \setminus \mathfrak{C}_M^0))$$

defined on three disjoint subspaces corresponding to a partition of the whole space  $\text{span}(\text{CB}(\mathfrak{C}_M))$ . Moreover, since even the images of these three operators are disjoint and cover the whole space  $\text{span}(\text{CB}(\mathfrak{C}_M))$ , their sum

$$W_M = W_{0,M} + W_{s,M} + W_{t,M}$$

defines an automorphism on the linear space of q-configurations

$$W_M : \text{span}(\text{CB}(\mathfrak{C})) \rightarrow \text{span}(\text{CB}(\mathfrak{C}))$$

By completion,  $W_M$  extends in a unique way to an operator on the Hilbert space of q-configurations.

**Definition 6** (time evolution operator). *The time evolution operator of  $M$  is the unique extension*

$$U_M : \ell^2(\mathfrak{C}_M) \rightarrow \ell^2(\mathfrak{C}_M)$$

of the linear operator  $W_M : \text{span}(\text{CB}(\mathfrak{C})) \rightarrow \text{span}(\text{CB}(\mathfrak{C}))$ .

As for the operator  $W_M$ , the time evolution operator  $U_M$  can be decomposed into three operators

$$\begin{aligned} U_{0,M} &: \ell^2(\mathfrak{C}_M^0 \setminus \mathfrak{C}_M^t) \rightarrow \ell^2(\mathfrak{C}_M^0 \setminus \mathfrak{C}_M^s) \\ U_{s,M} &: \ell^2(\mathfrak{C}_M^s \setminus \mathfrak{C}_M^0) \rightarrow \ell^2(\mathfrak{C}_M^s) \\ U_{t,M} &: \ell^2(\mathfrak{C}_M^t) \rightarrow \ell^2(\mathfrak{C}_M^t \setminus \mathfrak{C}_M^0) \end{aligned}$$

s.t.

$$U_M = U_{0,M} + U_{s,M} + U_{t,M}$$

**Definition 7** (QTM). *A pQTM is a Quantum Turing Machine (QTM) when its time evolution operator  $U_M$  is unitary.*

**2.9. Computations.** A computation of a QTM is an iteration of its evolution operator on some q-configuration. Since the time evolution operator of a QTM is unitary, it preserves the norm of its argument and maps q-configurations into q-configurations. By the way, this holds for the inverse  $U_M^{-1}$  of the time evolution operator too.

**Fact 8.** *Let  $M$  be a QTM. If  $|\phi\rangle \in \ell_1^2(\mathfrak{C}_M)$ , then  $U_M^i |\phi\rangle \in \ell_1^2(\mathfrak{C}_M)$ , for every  $i \in \mathbb{Z}$ .*

**Definition 9** (initial and final configurations). *A q-configuration  $|\phi\rangle$  is initial when  $|\phi\rangle \in \ell_1^2(\mathfrak{C}_M^{\text{init}})$  and is final when  $|\phi\rangle \in \ell_1^2(\mathfrak{C}_M^{\text{fin}})$ . By  $|\underline{n}\rangle$  we denote the initial configuration  $\langle \lambda, q_0, \underline{n}, 0 \rangle \in \mathfrak{C}_M^{\text{init}}$ .*

**Definition 10** (computations). *Let  $M$  be a QTM and let  $U_M$  be its time evolution operator. For an initial q-configuration  $|\phi\rangle \in \ell_1^2(\mathfrak{C}_M^{\text{init}})$ , the computation of  $M$  on  $|\phi\rangle$  is the denumerable sequence  $\{|\phi_i\rangle\}_{i \in \mathbb{N}}$  s.t.*

- (1)  $|\phi_0\rangle = |\phi\rangle$ ;
- (2)  $|\phi_i\rangle = U_M^i |\phi\rangle$ .

Clearly, any computation of a QTM  $M$  is univocally determined by its initial q-configuration. The computation of  $M$  on the initial q-configuration  $|\phi\rangle$  will be denoted by  $K_{|\phi\rangle}^M$ .

*Remark 11.* The definition of time evolution operator ensures that the final configurations reached along a computation are stable and do not interfere with other branches of the computation in superposition, which may enter into a final configuration later. Indeed, given a configuration  $|\phi\rangle = |\phi_f\rangle + |\phi_{nf}\rangle$ , where  $|\phi_f\rangle \in \ell^2(\mathfrak{C}_M^{\text{fin}})$  and  $|\phi_{nf}\rangle$  does not contain any final configuration, let  $\psi = U^i |\phi\rangle = U^i |\phi_f\rangle + U^i |\phi_{nf}\rangle$ . Any final configuration in  $U^i |\phi_{nf}\rangle$  has a value of the counter less than  $i$ , while any final configuration  $|\mathbf{C}, k\rangle \in |\phi\rangle$  formed of a plain configuration  $C$  and a counter  $k$  is mapped into a configuration  $|\mathbf{C}, i+k\rangle \in \psi$ . Moreover,  $|\mathbf{C}, k\rangle$  and  $|\mathbf{C}, i+k\rangle$  have the same coefficient in  $|\phi\rangle$  and  $|\psi\rangle$ , respectively, since  $\langle \psi | \mathbf{C}, i+k \rangle = \langle U^i |\phi\rangle, U^i |\mathbf{C}, k \rangle \rangle = \langle \phi | \mathbf{C}, k \rangle$ .

**2.10. Local conditions for unitary evolution.** In analogy of the main approaches in literature [3, 23], it is possible to state a set of local condition for the quantum transition function  $\delta_0$  of pQTM in order to ensure that the time evolution operator is unitary.

**Theorem 12.** *Let  $M$  be a pQTM with quantum transition function  $\delta_0$ . The time evolution operator  $U_M : \ell^2(\mathfrak{C}_M) \rightarrow \ell^2(\mathfrak{C}_M)$  of  $M$  is unitary iff  $\delta_0$  satisfies the local conditions:*

(1) for any  $(q, a) \in (\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma$

$$\sum_{(p,b,d) \in (\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D}} |\delta_0(q, a)(p, b, d)|^2 = 1$$

(2) for any  $(q, a), (q', a') \in (\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma$  with  $(q, a) \neq (q', a')$

$$\sum_{(p,b,d) \in (\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D}} \delta_0(q', a')(p, b, d)^* \delta_0(q, a)(p, b, d) = 0$$

(3) for any  $(q, a, b), (q', a', b') \in (\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma^2$

$$\sum_{p \in (\mathcal{Q} \setminus \mathcal{Q}_s)} \delta_0(q', a')(p, b', L)^* \delta_0(q, a)(p, b, R) = 0$$

The proof of the above theorem follows the main steps of the proofs given in the literature for B&V QTMs [3, 23]. First of all, one proves that the time evolution operator  $U_M$  is an isometry of  $\ell^2(\mathfrak{C})$ , and then that  $U_M$  is surjective. The first step is rather straightforward, since one can easily find the adjoint  $U_M^*$  of  $U_M$  and prove that it is a left-inverse of  $U_M$  iff the local conditions hold (we recall that an operator  $U$  is an isometry iff  $U^*U = 1$ , where  $U^*$  is the adjoint of  $U$ ). The second step is the hard part of the proof. Both [3] and [23] provide involved proofs of the fact that  $U^*$  is surjective. A previous version of the present paper included a direct and much simpler proof of Theorem 12—instead of proving that  $U_M^*$  is surjective, we show that  $U_M^*$  is the right inverse of  $U_M$ . We omit here the details of the proof of Theorem 12, and we refer the interested reader to the above mentioned version of the paper<sup>4</sup> available on ArXiv [12].

### 2.11. A comparison with Bernstein and Vazirani's QTMs: part 1.

We refer to B&V for the precise definitions of the QTMs used in that paper. For the sake of readability, we informally recall the notion of what they call *well formed, stationary, normal form QTMs* (B&V-QTMs in the following).

A B&V-QTM  $M = \langle \Sigma, \mathcal{Q}, \delta, q_0, q_f \rangle$  is defined as our QTM (with one source state and one target state only) with the following differences:

(1) the set of configurations coincides with all possible classical configurations, namely all the set  $\Sigma^* \times \mathcal{Q} \times \Sigma^*$ .

---

<sup>4</sup>The definition of QTM in [12] is slightly different then the one given here. In particular, it differs for the evolution in source and target states. However, the definition of QTM in [12] can be just seen as a particular implementation of the one given here (see Appendix B), and the proofs in the appendix of [12] easily adapt.

- (2) no superposition is allowed in the initial q-configuration (it must be a classical configuration  $\langle \alpha, q, \beta \rangle$  with amplitude 1);
- (3) let  $|C\rangle$  be such an initial configuration and let
 
$$k = \min\{j \mid U_M^j |C\rangle \text{ contains a final configuration}\}$$
 If such a  $k$  exists, then (i) all the configurations in  $U_M^k |C\rangle$  are final; (ii) for all  $i < k$ ,  $U_M^i |C\rangle$  does not contain any final configuration. We say in this case that the QTM halts in  $k$  steps in  $U_M^k |C\rangle$ ;
- (4) if a QTM halts, then the tape head is on the start cell of the initial configuration;
- (5) there is no counter and the transitions out of the final state or into the initial state are replaced by loops from the final state into the initial state, that is,  $\delta(q_f, a)(q_0, a, R) = 1$  for every  $a \in \Sigma$ . Therefore, because of the local unitary conditions, that must hold in the final state also, these are the only outgoing transitions from the final state, and the only incoming ones into the initial state, that is,  $\delta(q_f, a)(q', a', d) = 0$  if  $(q', a', d) \neq (q_0, a, R)$  and  $\delta(q', a')(q_0, a, d) = 0$  if  $(q', a', d) \neq (q_f, a, R)$ .

**Theorem 13.** *For any B&V-QTM  $M$  there is a QTM  $M'$  s.t. for each initial configuration  $|C\rangle$ , if  $M$  with input  $|C\rangle$  halts in  $k$  steps in a final configuration  $|\phi\rangle = U_M^k |C\rangle$ , then  $U_{M'}^k |C\rangle = |\phi\rangle$ .*

*Proof.* The QTM  $M'$  has the same states of  $M$ , the initial state  $q_0$  is its only source state, and the final state  $q_f$  is its only target state. Therefore, if  $M = \langle \Sigma, \mathcal{Q}, \delta, q_0, q_f \rangle$ , we take  $M' = \langle \Sigma, \mathcal{Q}, \{\delta_0\}, \{q_f\}, \delta_0, q_0, q_f \rangle$ , where  $\delta_0$  is the restriction of  $\delta$  to  $((\mathcal{Q} \setminus \mathcal{Q}_t) \times \Sigma) \rightarrow \ell^2((\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D})$ , that is, for each  $q \neq q_f$  and  $a \in \Sigma$ , we have  $\delta_0(q, a)(p, b, d) = \delta(q, a)(p, b, d)$ , for every  $(p, b, d) \in (\mathcal{Q} \setminus \mathcal{Q}_s) \times \Sigma \times \mathbb{D}$ . Since the local unitary conditions hold for  $\delta$  and, in a B&V-QTM,  $\delta(q, a)(q_0, b, d) = 0$ , when  $q \neq q_f$ , the unitary local conditions hold for  $\delta_0$  too.

By construction, if  $U_M^i |C\rangle$  is not final for  $0 \leq i < k$ , then  $|\phi_k\rangle = U_M^k |C\rangle = U_{M'}^k |C\rangle$ . In particular, this holds when  $|\phi_k\rangle$  is the final configuration of the B&V-QTM  $M$ .  $\square$

### 3. QUANTUM COMPUTABLE FUNCTIONS

In this section we address the problem of defining the concept of quantum computable function in an “ideal” way, without taking into account any measurement protocol. The problem of the observation protocol will be addressed in Section 4. Here we show how each QTM naturally defines a computable function from the sphere of radius 1 in  $\ell^2(\mathfrak{C}_M)$  to the set of (partial) probability distributions on the set of natural numbers.

#### 3.1. Probability distributions.

**Definition 14** (Probability distributions).

- (1) A partial probability distribution (PPD) of natural numbers is a function  $\mathbf{P} : \mathbb{N} \rightarrow \mathbb{R}_{[0,1]}$  such that  $\sum_{n \in \mathbb{N}} \mathbf{P}(n) \leq 1$ .
- (2) If  $\sum_{n \in \mathbb{N}} \mathbf{P}(n) = 1$ ,  $\mathbf{P}$  is a probability distribution (PD).
- (3)  $\mathbb{P}$  and  $\mathbb{P}_1$  denote the sets of all the PPDs and PDs, respectively.
- (4) If the set  $\{n : \mathbf{P}(n) \neq 0\}$  is finite,  $\mathbf{P}$  is finite.
- (5) Let  $\mathbf{P}', \mathbf{P}''$  be two PPDs, we say that  $\mathbf{P}' \leq \mathbf{P}''$  ( $\mathbf{P}' < \mathbf{P}''$ ) iff for each  $n \in \mathbb{N}$ ,  $\mathbf{P}'(n) \leq \mathbf{P}''(n)$  ( $\mathbf{P}'(n) < \mathbf{P}''(n)$ ).
- (6) Let  $\mathcal{P} = \{\mathbf{P}_i\}_{i \in \mathbb{N}}$  be a denumerable sequence of PPDs;  $\mathcal{P}$  is monotone iff  $\mathbf{P}_i \leq \mathbf{P}_j$ , for each  $i < j$ .

*Remark 15.* In the following, we shall also use the notation  $\mathbf{P}(\perp) = 1 - \sum_{n \in \mathbb{N}} \mathbf{P}(n)$ . By definition,  $0 \leq \mathbf{P}(\perp) \leq 1$ , and a PPD is a PD iff  $\mathbf{P}(\perp) = 0$ . We also stress that  $\leq$  is a partial order of PPDs and that any PD  $\mathbf{P}$  is maximal w.r.t. to  $\leq$ , since  $\mathbf{P} \leq \mathbf{P}'$  iff  $\mathbf{P} = \mathbf{P}'$ , for any PPD  $\mathbf{P}'$ .

**Definition 16** (limit of a sequence of PPDs). Let  $\mathcal{P} = \{\mathbf{P}_i\}_{i \in \mathbb{N}}$  be a sequence of PPDs. If for each  $n \in \mathbb{N}$  there exists  $l_n = \lim_{i \rightarrow \infty} \mathbf{P}_i(n)$ , we say that  $\lim_{i \rightarrow \infty} \mathbf{P}_i = \mathbf{P} : \mathbb{N} \rightarrow \mathbb{R}_{[0,1]}$ , with  $\mathbf{P}(n) = l_n$ .

### 3.2. Monotone sequences of probability distributions.

**Proposition 17.** Let  $\mathcal{P} = \{\mathbf{P}_i\}_{i \in \mathbb{N}} \subseteq \mathbb{P}$  be a monotone sequence of PPDs.

- (1)  $\lim_{i \rightarrow \infty} \mathbf{P}_i$  exists and it is the supremum  $\sqcup \mathcal{P}$  of  $\mathcal{P}$  ;
- (2)  $\sum_{n \in \mathbb{N}} (\sqcup \mathcal{P})(n) = \sqcup \left\{ \sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}}$ ;
- (3)  $\sqcup \mathcal{P} \in \mathbb{P}$ .

*Proof.* Since  $\mathbf{P}_i(n) \leq 1$ , every non-decreasing sequence  $\{\mathbf{P}_i(n)\}_{i \in \mathbb{N}}$  has a supremum  $\sup\{\mathbf{P}_i(n)\}_{i \in \mathbb{N}} = \lim_{i \rightarrow \infty} \mathbf{P}_i(n)$ . Thus,  $\lim_{i \rightarrow \infty} \mathbf{P}_i$  is defined and  $\mathbf{P}_i \leq \lim_{i \rightarrow \infty} \mathbf{P}_i$ , for every  $i \in \mathbb{N}$ . On the other hand, for any  $\mathbf{P}'$  s.t.  $\mathbf{P}_i \leq \mathbf{P}'$  for  $i \in \mathbb{N}$ , we have  $\lim_{i \rightarrow \infty} \mathbf{P}_i(n) = \sup\{\mathbf{P}_i(n)\}_{i \in \mathbb{N}} \leq \mathbf{P}'(n)$ , for every  $n \in \mathbb{N}$ ; namely,  $\lim_{i \rightarrow \infty} \mathbf{P}_i \leq \mathbf{P}'$ . We can then conclude (item 1) that  $\sqcup \mathcal{P} = \lim_{i \rightarrow \infty} \mathbf{P}_i$ .

Let us now prove item 2. First of all, since  $0 \leq \mathbf{P}_i(n) \leq (\sqcup \mathcal{P})(n)$  for  $i, n \in \mathbb{N}$ , we have  $\sum_{n \in \mathbb{N}} (\sqcup \mathcal{P})(n) \geq \sup \left\{ \sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}}$  and

$$\begin{aligned} \sup \left\{ \sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}} &\geq \sup \left\{ \sum_{n \leq k} \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}} \\ &= \sum_{n \leq k} \sup \left\{ \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}} = \sum_{n \leq k} (\sqcup \mathcal{P})(n) \end{aligned}$$

for any  $k \in \mathbb{N}$ . Thus,

$$\begin{aligned} \sup \left\{ \sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}} &\geq \sup \left\{ \sum_{n \leq k} (\sqcup \mathcal{P})(n) \right\}_{k \in \mathbb{N}} \\ &= \lim_{k \rightarrow \infty} \sum_{n \leq k} (\sqcup \mathcal{P})(n) = \sum_{n \in \mathbb{N}} (\sqcup \mathcal{P})(n) \end{aligned}$$



Summing up, we can conclude, as

$$\sum_{n \in \mathbb{N}} (\bigsqcup \mathcal{P})(n) \geq \sup \left\{ \sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \right\}_{i \in \mathbb{N}} \geq \sum_{n \in \mathbb{N}} (\bigsqcup \mathcal{P})(n)$$

Finally, by hypothesis,  $\sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \leq 1$ , for any  $i \in \mathbb{N}$ . Therefore,  $\sum_{n \in \mathbb{N}} (\bigsqcup \mathcal{P})(n) = \sup \{ \sum_{n \in \mathbb{N}} \mathbf{P}_i(n) \}_{i \in \mathbb{N}} \leq 1$ . Which proves (item 3) that  $\bigsqcup \mathcal{P} \in \mathbb{P}$ .  $\square$

**3.3. PPD sequence of a computation.** The computed output of a QTM will be defined (Definition 22) as the limit of the sequence of PPDs obtained along its computations.

**Definition 18** (probability and q-configurations). *Given a configuration  $C = \langle \alpha, q, \beta, n \rangle$ , let  $\mathbf{val}[C]$  be the number of 1 in  $\alpha\beta$ . For any  $|\phi\rangle \in \ell^2(\mathfrak{C})$ , let us define  $\mathbf{P}_{|\phi\rangle} : \mathbb{N} \rightarrow \mathbb{R}_{\geq 0}$  s.t.*

$$\mathbf{P}_{|\phi\rangle}(n) = \sum_{C \in \mathfrak{C}^{fin}, \mathbf{val}[C]=n} |e_C|^2$$

when  $|\phi\rangle = \sum_{C \in \mathfrak{C}} e_C |C\rangle$ .

**Proposition 19.** *If  $|\phi\rangle$  is a q-configuration,  $\mathbf{P}_{|\phi\rangle}$  is a PPD. Moreover, it is a PD iff  $|\phi\rangle$  is final.*

*Proof.* Let  $|\phi\rangle = |\phi_0\rangle + |\phi_f\rangle$  with  $|\phi_0\rangle \in \ell^2(\mathfrak{C}_M \setminus \mathfrak{C}_M^{fin})$ , and  $|\phi_f\rangle \in \ell^2(\mathfrak{C}_M^{fin})$ . By the definition of  $\mathbf{P}_{|\phi\rangle}$ , it is readily seen that  $\mathbf{P}_{|\phi\rangle} = \mathbf{P}_{|\phi_f\rangle}$  and that  $\sum_{n \in \mathbb{N}} \mathbf{P}_{|\phi\rangle}(n) = \sum_{n \in \mathbb{N}} \mathbf{P}_{|\phi_f\rangle}(n) = \|\phi_f\| \leq 1$ . Therefore,  $\mathbf{P}_{|\phi\rangle}$  is a PPD. Moreover, since  $\|\phi_f\| = 1$  iff  $\|\phi_0\| = 0$ . We see that  $\mathbf{P}_{|\phi\rangle}$  is a PD iff  $|\phi_0\rangle = 0$ , that is, iff  $|\phi\rangle = |\phi_f\rangle$ .  $\square$

**Definition 20.** *For any q-configuration  $|\phi\rangle$ , we shall say that  $\mathbf{P}_{|\phi\rangle}$  is the PPD associated to  $|\phi\rangle$ , and we shall denote by  $\mathbf{P}_{K_{|\phi\rangle}^M}$  the sequence of PPDs  $\{\mathbf{P}_{|\phi_i\rangle}\}_{i \in \mathbb{N}}$  associated to the computation  $K_{|\phi\rangle}^M = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$ .*

The PPD sequence of any QTM computation is monotone. In the simple proof of this key property (Theorem 21) we see at work all the constraints on the time evolution of a QTM  $M$ .

- (1) That when in a final (target) configuration, the machine can only increment the counter; as a consequence, the **val** of final (target) configurations does not change.
- (2) That when entering for the first time into a final (target) state, the value of the counter is initialised to 0.
- (3) That when in a final (target) configuration  $|\phi, n\rangle$ , the counter gives the number of steps  $n$  since  $M$  is looping into the plain configuration  $\phi$ .

We stress that the last two properties defuse quantum interference between final configurations reached in a different number of steps.

**Theorem 21** (monotonicity of computations). *For any computation  $K_{|\phi\rangle}^M$  of a QTM  $M$ , the sequence of PPDs  $\mathbf{P}_{K_{|\phi\rangle}^M}$  is monotone.*

*Proof.* Let us prove that  $\mathbf{P}_{|\phi\rangle} \leq \mathbf{P}_{U|\phi\rangle}$ , for every  $|\phi\rangle \in \ell^2(\mathfrak{C}_M)$ . Let  $|\phi\rangle = |\phi_0\rangle + \sum_{k \in \mathbb{N}} |\phi_{f,k}\rangle$ , with  $|\phi_0\rangle \in \ell^2(\mathfrak{C}_M \setminus \mathfrak{C}_M^{fin})$ , and  $|\phi_{f,k}\rangle \in \ell^2(\mathfrak{C}_M^{fin} \cap \mathfrak{C}_M^k)$ , where  $\mathfrak{C}_M^k$  is the set of the configurations of  $M$  whose counter is equal to  $k \in \mathbb{N}$ . For every  $n \in \mathbb{N}$ , we see that  $\mathbf{P}_{|\phi\rangle}(n) = \sum_{k \in \mathbb{N}} \mathbf{P}_{|\phi_{f,k}\rangle}(n)$ . Let  $|\psi\rangle = |\psi_0\rangle + \sum_{k \in \mathbb{N}} |\psi_{f,k}\rangle = U|\phi\rangle$ . By the definition of  $U$ , we see that  $U|\phi_0\rangle = |\psi_0\rangle + |\psi_{f,0}\rangle$  and  $U|\phi_{f,k}\rangle = |\psi_{f,k+1}\rangle$ . Therefore,  $\mathbf{P}_{|\psi\rangle}(n) = \sum_{k \in \mathbb{N}} \mathbf{P}_{|\psi_{f,k}\rangle}(n) = \mathbf{P}_{|\psi_{f,0}\rangle}(n) + \sum_{k \in \mathbb{N}} \mathbf{P}_{|\phi_{f,k}\rangle}(n) = \mathbf{P}_{|\psi_{f,0}\rangle}(n) + \mathbf{P}_{|\phi\rangle}(n) \geq \mathbf{P}_{|\phi\rangle}(n)$ .  $\square$

**3.4. Computed output.** We can now come back to the definition of the computed output of a QTM computation. The easy case is when a computation reaches a final q-configuration  $|\psi\rangle \in \mathfrak{C}_M^{fin}$  (meaning that all the classical computations in superposition are “terminated”)—in this case the computed output is the PD  $\mathbf{P}_{|\psi\rangle}$ . The QTM keeps computing and transforming  $|\psi\rangle$  into other configurations, but all these configurations have the same PD. However, we want to give meaning also to “infinite” computations, which never reach a final q-configuration, yet producing some final configurations in the superpositions. For this purpose, we define the computed output as the limit of the PPDs yielded by the computation.

**Definition 22** (computed output of a QTM). *Let  $K_{|\phi\rangle}^M = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$  be the computation of the QTM  $M$  on the initial q-configuration  $|\phi\rangle$ . The computed output of  $M$  on the initial q-configuration  $|\phi\rangle$  is the PPD  $\mathbf{P} = \lim_{i \rightarrow \infty} \mathbf{P}_{|\phi_i\rangle}$ , which we shall also denote by  $\lim K_{|\phi\rangle}^M$ , or by the notation  $M_{|\phi\rangle} \rightarrow \mathbf{P}$ .*

Let us remark that, by Proposition 17 and Theorem 21, the limit in the above definition is well-defined for any computation. Therefore, a QTM has a computed output for any initial q-configuration  $|\phi\rangle$ .

**Definition 23** (finitary computations). *Given a QTM  $M$ , a q-configuration  $|\phi\rangle$  is finitary if it is an element of  $\text{span}(\text{CB}(\mathfrak{C}_M))$ . A computation  $K_{|\phi\rangle}^M = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$  is finitary with computed output  $\mathbf{P}$  if there exists a  $k$  s.t.  $|\phi_k\rangle$  is final and  $\mathbf{P}_{|\phi_k\rangle} = \mathbf{P}$ .*

**Proposition 24.** *Let  $K_{|\phi\rangle}^M = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$  be a finitary computation with computed output  $\mathbf{P}$ , that is,  $M_{|\phi\rangle} \rightarrow \mathbf{P}$ .*

- (1) *There exists a  $k$ , such that for each  $j \geq k$ ,  $|\phi_j\rangle$  is final and  $\mathbf{P}_{|\phi_j\rangle} = \mathbf{P}$ .*
- (2)  *$\mathbf{P}$  is a PD.*

*Proof.* By definition, there is a  $k$  s.t.  $|\phi_k\rangle$  is final. Let  $\mathbf{P}_{|\phi_k\rangle} = \mathbf{P}$ . By Proposition 19,  $\mathbf{P}$  is a PD. By monotonicity,  $\mathbf{P} \leq \mathbf{P}_{|\phi_j\rangle}$ , for every  $j \geq k$ . Thus, since any PD is maximal for  $\leq$  (see Remark 15),  $\mathbf{P} = \mathbf{P}_{|\phi_j\rangle}$ , for every  $j \geq k$ .  $\square$

Given a computation  $K_{|\phi\rangle}^M$ , we can then distinguish the following cases:

- (1)  $K_{|\phi\rangle}^M$  is finitary. In this case  $M_{|\phi\rangle} \rightarrow \mathbf{P} \in \mathbb{P}_1$ ; the output of the computation is then a PD and is determined after a finite number of steps;
- (2)  $K_{|\phi\rangle}^M$  is not finitary, but  $M_{|\phi\rangle} \rightarrow \mathbf{P} \in \mathbb{P}_1$ . The output is a PD and is determined as a limit;
- (3)  $K_{|\phi\rangle}^M$  is not finitary, and  $M_{|\phi\rangle} \rightarrow \mathbf{P} \in \mathbb{P} \setminus \mathbb{P}_1$  (the sum of the probabilities of observing natural numbers is  $p < 1$ ). Not only the result is determined as a limit, but we cannot extract a PD from the output.

The first two cases above give rise to what Definition 25 calls a q-total function. Observe, however, that for an external observer, cases (2) and (3) are in general indistinguishable, since at any finite stage of the computation we may observe only a finite part of the computed output.

For some examples of QTMs and their computed output, see Section 5.

**3.5. Quantum partial computable functions.** We want our quantum computable functions to be defined over a natural extension of the natural numbers. Recall that, for any  $n \in \mathbb{N}$ ,  $\underline{n}$  denotes the string  $1^{n+1}$  and that  $|\underline{n}\rangle = |\langle \lambda, \mathbf{q}_0, \underline{n}, 0 \rangle\rangle$ . When using a QTM for computing a function, we stipulate that initial q-configurations are superpositions of initial classical configurations of the shape  $|\underline{n}\rangle$ . Such q-configurations are naturally isomorphic to the space  $\ell_1^2 = \{\phi : \mathbb{N} \rightarrow \mathbb{C} \mid \sum_{n \in \mathbb{N}} |\phi(n)|^2 = 1\}$  of square summable, denumerable sequences with unitary norm, under the bijective mapping  $\nu(\sum d_k n_k) = \sum d_k |\underline{n}_k\rangle$ .

**Definition 25** (partial quantum computable functions).

- (1) A function  $f : \ell_1^2 \rightarrow \mathbb{P}$  is partial quantum computable (*q-computable*) if there exists a QTM  $M$  s.t.  $f(\mathbf{x}) = \mathbf{P}$  iff  $M_{\nu(\mathbf{x})} \rightarrow \mathbf{P}$ .
- (2) A q-partial computable function  $f$  is quantum total (*q-total*) if for each  $\mathbf{x}$ ,  $f(\mathbf{x}) \in \mathbb{P}_1$ .

$\mathcal{QCF}$  is the class of partial quantum computable functions.

#### 4. OBSERVABLES

While the evolution of a closed quantum system (e.g., a QTM) is reversible and deterministic once its evolution operator is known, a (global) measurement of a q-configuration is an irreversible process, which causes the collapse of the quantum state to a new state. Technically, a measurement corresponds to a projection on a subspace of the Hilbert space of quantum states. For the sake of simplicity, in the case of QTMs, let us restrict to measurements observing if a configuration belongs to the subspace described by some set of base configurations  $\mathcal{B}$ . The effect of such a measurement is summarised by the following:

### Measurement postulate

Given a set of configurations  $\mathcal{B} \subseteq \mathfrak{C}$ , a measurement observing if a quantum configuration  $|\phi\rangle = \sum_{C \in \mathfrak{C}} e_C |C\rangle$  belongs to the subspace generated by  $\text{CB}(\mathcal{B})$  gives a positive answer with a probability  $p = \sum_{C \in \mathcal{B}} |e_C|^2$ , equal to the square of the norm of the projection of  $|\phi\rangle$  onto  $\ell^2(\mathcal{B})$ , causing at the same time a collapse of the configuration into the normalised projection  $\sum_{C \in \mathcal{B}} p^{-\frac{1}{2}} e_C |C\rangle$ ; dually, with probability  $1-p = \sum_{C \notin \mathcal{B}} |e_C|^2$ , it gives a negative answer and a collapse onto the subspace  $\ell^2(\mathfrak{C} \setminus \mathcal{B})$  orthonormal to  $\ell^2(\mathcal{B})$ , that is, into the normalised configuration  $\sum_{C \notin \mathcal{B}} (1-p)^{-\frac{1}{2}} e_C |C\rangle$ .

Because of the *irreversible* modification produced by any measurement on the current configuration, and therefore on the rest of the computation, we must deal with the problem of how to read the result of a computation. In other words, we need to establish some protocol to observe when a QTM has eventually reached a final configuration, and to read the corresponding result.

**4.1. The approach of Bernstein and Vazirani.** We already discussed how B&V's "sensible" QTMs are machines where all the computations in superposition are in some sense terminating, and reach the final state at the same time (are "stationary", in their terminology). More precisely, Definition 3.11 of B&V reads: "*A final configuration of a QTM is any configuration in [final] state. If when QTM  $M$  is run with input  $x$ , at time  $T$  the superposition contains only final configurations, and at any time less than  $T$  the superposition contains no final configuration, then  $M$  halts with running time  $T$  on input  $x$ .*"

This is a good definition for a theory of computational complexity (where the problems are classical, and the inputs of QTMs are always classical) but it is of little use for developing a theory of effective quantum functions. Indeed, inputs of a B&V-QTM *must* be classical—we cannot extend by linearity a B&V-QTM on inputs in  $\ell_1^2$ , since there is no guarantee whatsoever that on different inputs the same QTM halts with the same running time.

**4.2. The approach of Deutsch.** Deutsch [10] assumes that QTMs are enriched with a termination bit  $T$ . At the beginning of a computation,  $T$  is set to 0, then, the machine sets this termination bit to 1 when it enters into a final configuration. If we write  $|\mathsf{T} = i\rangle$  for the function that returns 1 when the termination bit is set to  $i$ , and 0 otherwise, a generic q-configuration of a Deutsch's QTM can be written as

$$|\phi\rangle^2 = |\mathsf{T} = 0\rangle \otimes \sum_{C \notin \mathfrak{C}^{fin}} e_C |C\rangle + |\mathsf{T} = 1\rangle \otimes \sum_{D \in \mathfrak{C}^{fin}} d_D |D\rangle$$

The observer periodically measures  $T$  in a non destructive way (that is, without modifying the rest of the state of the machine).

- (1) If the result of the measurement of  $T$  gives the value 0,  $|\phi\rangle$  collapses (with a probability equal to  $\sum_{C \notin \mathfrak{C}^{fin}} |e_C|^2$ ) to the q-configuration

$$|\psi'\rangle = \frac{|\mathbf{T} = 0\rangle \otimes \sum_{C \notin \mathfrak{C}^{fin}} e_C |C\rangle}{\sqrt{\sum_{C \notin \mathfrak{C}^{fin}} |e_C|^2}}$$

and the computation continues with  $|\psi'\rangle$ .

- (2) If the result of the measurement of  $T$  gives the value 1,  $|\phi\rangle$  collapses (with probability  $\sum_{D \in \mathfrak{C}^{fin}} |d_D|^2$ ) to

$$|\psi''\rangle = \frac{|\mathbf{T} = 1\rangle \otimes \sum_{D \in \mathfrak{C}^{fin}} d_D |D\rangle}{\sqrt{\sum_{D \in \mathfrak{C}^{fin}} |d_D|^2}}$$

and, immediately after the collapse, the observer makes a further measurement of the component  $\frac{\sum_{D \in \mathfrak{C}^{fin}} d_D |D\rangle}{\sqrt{\sum_{D \in \mathfrak{C}^{fin}} |d_D|^2}}$  in order to read-back a final configuration.

Note that Deutsch's protocol (in an irreversible way) spoils at each step the superposition of configurations. The main point of Deutsch's approach is that a measurement must be performed immediately after some computation enters into a final state. In fact, since at the following step the evolution might lead the machine to exit the final state modifying the content of the tape, we would not be able to measure at all this output. In other words, either the termination bit acts as a trigger that forces a measurement each time it is set, or we perform a measurement after each step of the computation.

**4.3. Our approach.** The measurement of the output computed by our QTMs can be performed by following a variant of Deutsch's approach. Because of the particular structure of the transition function of our QTMs, we shall see that we do not need any additional termination bit, that a measurement can be performed at any moment of the computation, and that indeed we can perform several measurements at distinct points of the computation without altering the result (in terms of the probabilistic distribution of the observed output).

Given a q-configuration  $|\phi\rangle = |\phi_f\rangle + |\phi_{nf}\rangle$ , where  $|\phi_f\rangle \in \ell^2(\mathfrak{C}^{fin})$  and  $|\phi_{nf}\rangle \in \ell^2(\mathfrak{C} \setminus \mathfrak{C}^{fin})$ , our output measurement tries to get an output value from  $|\phi\rangle$  by the following procedure:

- (1) first of all, we observe the final states of  $|\phi\rangle$ , forcing the q-configuration to collapse either into the final q-configuration  $|\phi_f\rangle / \|\phi_f\|$ , or into the q-configuration  $|\phi_{nf}\rangle / \|\phi_{nf}\|$ , which does not contain any final configuration;

- (2) then, if the q-configuration collapses into  $|\phi_f\rangle / \|\phi_f\rangle\|$ , we observe one of these configurations, say  $|C\rangle$ , which gives us the observed output  $\mathbf{val}[C] = n$ , forcing the q-configuration to collapse into the final base q-configuration  $(e_c/|e_c|)|C\rangle$ ;
- (3) otherwise, we leave unchanged the q-configuration  $|\phi_{nf}\rangle / \|\phi_{nf}\rangle\|$  obtained after the first observation, and we say that we have observed the special value  $\perp$ .

Summing up, an output measurement of  $|\phi\rangle$  may lead to observe an output value  $n \in \mathbb{N}$  associated to a collapse into a base final configuration  $|C\rangle \in |\phi\rangle$  s.t.  $\mathbf{val}[\phi] = n$  or to observe the special value  $\perp$  associated to a collapse into a q-configuration which does not contain any final configuration.

**Definition 26** (output observation). *An output observation with collapsed q-configuration  $|\psi\rangle$  and observed output  $x \in \mathbb{N} \cup \{\perp\}$  is the result of an output measurement of the q-configuration  $|\phi\rangle = \sum_{C \in \mathfrak{C}} e_C |C\rangle$ . Therefore, it is a triple  $|\phi\rangle \downarrow_x |\psi\rangle$  s.t.*

- (1) *either  $x = n \in \mathbb{N}$ , and*

$$|\psi\rangle = \frac{e_C}{|e_C|} |C\rangle \quad \text{with} \quad C \in \mathfrak{C}^{fin} \text{ and } \mathbf{val}[C] = n \text{ and } e_C \neq 0$$

- (2) *or  $x = \perp$ , and*

$$|\psi\rangle = \frac{|\phi_{nf}\rangle}{\|\phi_{nf}\rangle\|} \quad \text{where} \quad |\phi_{nf}\rangle = \sum_{C \notin \mathfrak{C}^{fin}} e_C |C\rangle \text{ and } |\phi_{nf}\rangle \neq 0$$

*The probability of an output observation is defined by*

$$\Pr\{|\phi\rangle \downarrow_x |\psi\rangle\} = \begin{cases} |e_C|^2 & \text{if } x = n \in \mathbb{N} \\ \|\phi_{nf}\rangle\|^2 & \text{if } x = \perp \end{cases}$$

*Remark 27.* Let  $e|C\rangle \downarrow_x |\phi\rangle$ , with  $C \in \mathfrak{C}^{fin}$  and  $\mathbf{val}[C] = n$ . Since  $e|C\rangle$  is a q-configuration,  $|e| = 1$ . By the definition of output observation,  $x = n$  and  $|\phi\rangle = (e/|e|)U|C\rangle = e|D\rangle$ , with  $|D\rangle = U|C\rangle \in \mathbf{CB}(\mathfrak{C}^{fin})$  and  $\mathbf{val}[D] = n$ . Moreover,  $\Pr\{e|C\rangle \downarrow_x |\phi\rangle\} = |e|^2 = 1$ .

*Remark 28.* For every pair  $|\phi\rangle \downarrow_{x_1} |\psi_1\rangle$  and  $|\phi\rangle \downarrow_{x_2} |\psi_2\rangle$  of distinct output observations,  $\psi_1$  and  $\psi_2$  are in the orthonormal subspaces generated by the two disjoint sets  $\mathcal{B}_1, \mathcal{B}_2 \subseteq \mathfrak{C}$ , where  $\mathcal{B}_i = \{C \in \mathfrak{C} \mid |C\rangle \in |\psi_i\rangle\}$ .

**Definition 29** (observed run). *Let  $M$  be a QTM and  $U_M$  be its time evolution operator. For any monotone increasing function  $\tau : \mathbb{N} \rightarrow \mathbb{N}$  (that is,  $\tau(i) < \tau(j)$  for  $i < j$ ):*

- (1) *a  $\tau$ -observed run of  $M$  on the initial q-configuration  $|\phi\rangle$  is a sequence  $\{|\phi_i\rangle\}_{i \in \mathbb{N}}$  s.t.:*
  - (a)  $|\phi_0\rangle = |\phi\rangle$ ;
  - (b)  $U_M |\phi_h\rangle \downarrow_{x_i} |\phi_{h+1}\rangle$ , when  $h = \tau(i)$  for some  $i \in \mathbb{N}$ ;
  - (c)  $|\phi_{h+1}\rangle = U_M |\phi_h\rangle$  otherwise.

- (2) A finite  $\tau$ -observed run of length  $k$  is any finite prefix of length  $k+1$  of some  $\tau$ -observed run. Notation: if  $R = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$ , then  $R[k] = \{|\phi_i\rangle\}_{i \leq k}$ .

*Remark 30.* We stress that, given a  $\tau$ -observed run  $R = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$ :

- (1) either it never obtains a value  $n \in \mathbb{N}$  as the result of an output observation, and then it never reaches a final configuration;
- (2) or it eventually obtains such a value collapsing the q-configuration into a base final configuration  $e|C\rangle$  s.t.  $|e| = 1$  and  $\mathbf{val}[C] = n$ , and from that point onward all the configurations of the run are base final configurations  $e|C_j\rangle = eU^j|C\rangle$  s.t.  $\mathbf{val}[C_j] = n$ , and all the following observed outputs are equal to  $n$  (see Remark 27).

**Definition 31.** Let  $R = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$  be a  $\tau$ -observed run.

- (1) The sequence  $\{x_i\}_{i \in \mathbb{N}}$  s.t.  $|\phi_h\rangle \downarrow_{x_i} |\phi_{h+1}\rangle$ , with  $h = \tau(i)$ , is the output sequence of the  $\tau$ -observed run  $R$ .
- (2) The observed output of  $R$  is the value  $x \in \mathbb{N} \cup \{\perp\}$  (notation:  $R \downarrow_x$ ) defined by:
  - (a)  $x = n \in \mathbb{N}$ , if  $x_i = n$  for some  $i \in \mathbb{N}$ ;
  - (b)  $x = \perp$  otherwise.
- (3) For any  $k$ , the output sequence of the finite  $\tau$ -observed run  $R[\tau(k)+1]$  is the finite sequence  $\{x_i\}_{i \leq k}$  and  $x_k$  is its observed output.

**Definition 32** (probability of a run). Let  $R = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$  be a  $\tau$ -observed run.

- (1) For  $k \in \mathbb{N}$ , the probability of the finite  $\tau$ -observed run  $R[k]$  is inductively defined by
  - (a)  $\Pr\{R[0]\} = 1$ ;
  - (b)  $\Pr\{R[k+1]\} = \begin{cases} \Pr\{R[k]\} \Pr\{|\phi_k\rangle \downarrow_{x_i} |\phi_{k+1}\rangle\} & \text{when } k = \tau(i) \\ & \text{for some } i \in \mathbb{N} \\ \Pr\{R[k]\} & \text{otherwise} \end{cases}$
- (2)  $\Pr\{R\} = \lim_{k \rightarrow \infty} \Pr\{R[k]\}$ .

We remark that  $\Pr\{R\}$  is well-defined, since  $1 \geq \Pr\{R[i]\} \geq \Pr\{R[j]\} > 0$ , for every  $i \leq j$ . Therefore,

$$1 \geq \Pr\{R\} = \lim_{k \rightarrow \infty} \Pr\{R[k]\} = \inf\{\Pr\{R[k]\}\}_{k \in \mathbb{N}} \geq 0.$$

*Remark 33.* Let  $R = \{|\phi_i\rangle\}_{i \in \mathbb{N}}$  be a  $\tau$ -observed run s.t.  $R \downarrow_n$ , for some  $n \in \mathbb{N}$ . As observed in Remark 30, for some  $k$ , we have  $R[\tau(k)] \downarrow_n$  and  $R[\tau(j)] \downarrow_\perp$ , for  $j < k$ ; moreover, for  $i > \tau(k)$ ,  $|\phi_i\rangle = e|C_i\rangle$  with  $|e| = 1$ ,  $C_i \in \mathfrak{C}^{fin}$ , and  $\mathbf{val}[C_i] = n$ . As a consequence,  $\Pr\{R[\tau(k)+1]\} = \Pr\{R[i]\}$ , for any  $i > \tau(k)$  (since, by Remark 27,  $\Pr\{|\phi_{\tau(i)}\rangle \downarrow_n |\phi_{\tau(i)+1}\rangle\} = 1$ ) and  $\Pr\{R\} = \lim_{i \rightarrow \infty} \Pr\{R[i]\} = \Pr\{R[\tau(k)+1]\}$ .

**Definition 34** (observed computation). *The  $\tau$ -observed computation of a QTM  $M$  on the initial  $q$ -configuration  $|\phi\rangle$ , is the set  $\mathcal{K}_{|\phi\rangle, \tau}^M$  of the  $\tau$ -observed runs of  $M$  on  $|\phi\rangle$  with the measure  $\Pr : \mathcal{P}(\mathcal{K}_{|\phi\rangle, \tau}^M) \rightarrow \mathbb{C}$  defined by*

$$\Pr \mathcal{B} = \sum_{R \in \mathcal{B}} \Pr\{R\}$$

for every  $\mathcal{B} \subseteq \mathcal{K}_{|\phi\rangle, \tau}^M$ .

By  $\mathcal{K}[k]_{|\phi\rangle, \tau}^M$  we shall denote the set of the finite  $\tau$ -observed runs of length  $k$  of  $M$  on  $|\phi\rangle$ , with the measure  $\Pr$  on its subsets (see Definition 34).

It is immediate to observe that the set  $\mathcal{K}_{|\phi\rangle, \tau}^M$  naturally defines an infinite tree labelled with  $q$ -configurations, where each infinite path starting from the root  $|\phi\rangle$  corresponds to a  $\tau$ -observed run in  $\mathcal{K}_{|\phi\rangle, \tau}^M$ .

**Lemma 35.** *Given  $R_1, R_2 \in \mathcal{K}_{|\phi\rangle, \tau}^M$ , with  $R_1 = \{\phi_{1,i}\}_{i \in \mathbb{N}} \neq \{\phi_{2,i}\}_{i \in \mathbb{N}} = R_2$ , there is  $k \geq 0$  s.t.*

- (1)  $\phi_{1,i} = \phi_{2,i}$  for  $i \leq \tau(k)$ , that is,  $R_1[\tau(k)] = R_2[\tau(k)]$ ;
- (2) for  $i > \tau(k)$ , the  $q$ -configurations  $\phi_{1,i} \neq \phi_{2,i}$  are in two orthonormal subspaces generated by two distinct subsets of  $\mathfrak{C}$ .

*Proof.* Let  $R_1[h] = R_2[h]$  be the longest common prefix of  $R_1$  and  $R_2$ . Since they both starts with  $|\phi\rangle$ , such a prefix is not empty; moreover, by the definition of  $\tau$ -observed run, it is readily seen that  $h = \tau(k)$ , for some  $k$ .

Let us now prove item 2. If we take  $\mathcal{B}_{a,j} = \{C \in \mathfrak{C} \mid |C\rangle \in |\psi_{a,h+1+j}\rangle\}$ , for  $a = 1, 2$  and  $j \in \mathbb{N}$ , we need to prove that  $\mathcal{B}_{1,j} \cap \mathcal{B}_{2,j} = \emptyset$ , for every  $j$ . By construction,  $\phi_{1,h+1} \neq \phi_{2,h+1}$  and, for  $a = 1, 2$ ,  $\phi_h \downarrow_{x_a} \phi_{a,h+1}$ , for some  $x_a$ , with  $\phi_h = \phi_{a,h}$ . Moreover,  $\mathcal{B}_{1,0} \cap \mathcal{B}_{2,0} = \emptyset$  (see Remark 28), at least one of the two observed values  $x_1, x_2$  is not  $\perp$ , and one of the two  $q$ -configurations  $|\phi_{1,h+1}\rangle, |\phi_{2,h+1}\rangle$  is a final base  $q$ -configuration. W.l.o.g., let us assume that  $x_1 \in \mathbb{N}$  and let  $|\phi_{1,h+1}\rangle = e |C, n\rangle$ , for some final plain configuration  $C$  and some  $n \in \mathbb{N}$ . Then,  $|\phi_{1,h+1+j}\rangle = e U^j |C, n\rangle = e |C, n+j\rangle$  (see Remark 30) and  $\mathcal{B}_{1,j} = \{|C, n+j\rangle\}$ , for  $j \in \mathbb{N}$ . Thus, to prove the assertion, it suffices to show that  $|C, n+j\rangle \notin \mathcal{B}_{2,j}$ , for  $j \in \mathbb{N}$ . Let us distinguish two cases:

- (1)  $x_2 \in \mathbb{N}$  and  $|\psi_{2,h+1}\rangle = e' |D, m\rangle$ , where  $D$  is a final plain configuration and  $m \in \mathbb{N}$ . As already seen for  $|\psi_{1,h+1+j}\rangle$ , for  $j \in \mathbb{N}$ , we have  $|\psi_{2,h+1+j}\rangle = e' U^j |D, m\rangle = e' |D, m+j\rangle$  and  $\mathcal{B}_{2,j} = \{|D, m+j\rangle\}$ . Therefore,  $|C, n+j\rangle \notin \mathcal{B}_{2,j}$ , since  $|C, n+j\rangle \neq |D, m+j\rangle$ , for  $|C, n\rangle \neq |D, m\rangle$  by construction.
- (2)  $|\psi_{2,h+1}\rangle \in \ell^2(\mathfrak{C} \setminus \mathfrak{C}^{fin})$ . Let  $|D, m\rangle \in \mathcal{B}_{2,j} \cap \mathfrak{C}^{fin}$ , where  $D$  is a plain configuration and  $m \in \mathbb{N}$ . By induction on  $j$ , it is readily seen that  $m < j \leq n+j$ . Thus,  $|D, m\rangle \neq |C, n+j\rangle$ .

□

**Lemma 36.** *Let  $K_{|\phi\rangle}^M = \{\phi_i\}_{i \in \mathbb{N}}$  be the computation of the QTM  $M$  on the initial  $q$ -configuration  $|\phi\rangle$  and  $\mathcal{K}_{|\phi\rangle, \tau}^M$  be the  $\tau$ -observed computation on the*



same initial configuration. For every  $k \in \mathbb{N}$ , we have that

$$|\phi_k\rangle = \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R\}} |\psi_R\rangle$$

where  $|\psi_R\rangle$  is the last  $q$ -configuration of the finite run  $R$  of length  $k$ .

*Proof.* By definition,  $\phi = \phi_0$  and  $R = \{\phi\}$  with  $\Pr\{R\} = 1$ , and  $\psi_R = \phi$  is the only run of length 0 in  $\mathcal{K}_{|\phi\rangle, \tau}^M$ . Therefore, the assertion trivially holds for  $k = 0$ .

Let us then prove the assertion by induction on  $k$ . By definition and the induction hypothesis

$$|\phi_{k+1}\rangle = U_M |\phi_k\rangle = \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R\}} U_M |\psi_R\rangle$$

We have two possibilities:

- (1)  $k \neq \tau(i)$ , for any  $i$ . In this case, there is a bijection between the runs of length  $k$  and those of length  $k+1$ , since each run  $R' \in \mathcal{K}[k+1]_{|\phi\rangle, \tau}^M$  is obtained from a run  $R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M$  with last  $q$ -configuration  $|\psi_R\rangle$ , by appending to  $R$  the  $q$ -configuration  $|\psi_{R'}\rangle = U_M |\psi_R\rangle$ . Moreover, since by definition,  $\Pr\{R'\} = \Pr\{R\}$ , we can conclude that

$$|\phi_{k+1}\rangle = \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R\}} U_M |\psi_R\rangle = \sum_{R' \in \mathcal{K}[k+1]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R'\}} |\psi_{R'}\rangle$$

- (2)  $k = \tau(i)$ , for some  $i$ . In this case, every  $R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M$  with last  $q$ -configuration  $|\psi_R\rangle$  generates a run  $R'$  of length  $k+1$  for every output observation  $|\psi_R\rangle \downarrow_x |\psi_{R'}\rangle$ , where  $R'$  is obtained by appending  $|\psi_{R'}\rangle$  to  $R$ . Therefore, let  $R = \{|\psi_i\rangle\}_{i \leq k} \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M$  and

$$\mathcal{B}_R = \{\{|\psi_i\rangle\}_{i \leq k+1} \mid |\psi_k\rangle \downarrow_x |\psi_{k+1}\rangle\}$$

By applying Definition 26, we easily check that

$$U_M |\psi_R\rangle = \sum_{R' \in \mathcal{B}_R} \sqrt{\Pr\{|\psi_R\rangle \downarrow_x |\psi_{R'}\rangle\}} |\psi_{R'}\rangle$$

Thus, by substitution, and  $\Pr\{R\} \Pr\{|\psi_R\rangle \downarrow_x |\psi_{R'}\rangle\} = \Pr\{R'\}$

$$\begin{aligned}
|\phi_{k+1}\rangle &= \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R\}} U_M |\psi_R\rangle \\
&= \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sum_{R' \in \mathcal{B}_R} \sqrt{\Pr\{R\}} \sqrt{\Pr\{|\psi_R\rangle \downarrow_x |\psi_{R'}\rangle\}} |\psi_{R'}\rangle \\
&= \sum_{R' \in \bigcup_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \mathcal{B}_R} \sqrt{\Pr\{R'\}} |\psi_{R'}\rangle \\
&= \sum_{R' \in \mathcal{K}[k+1]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R'\}} |\psi_{R'}\rangle
\end{aligned}$$

since  $\bigcup_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \mathcal{B}_R = \mathcal{K}[k+1]_{|\phi\rangle, \tau}^M$ .

□

We are finally in the position to prove that our observation protocol is compatible with the probability distributions that we defined as computed output of a QTM computation.

**Theorem 37.** *Let  $K_{|\phi\rangle}^M = \{\phi_i\}_{i \in \mathbb{N}}$  be the computation of the QTM  $M$  on the initial  $q$ -configuration  $|\phi\rangle$  and  $\mathcal{K}_{|\phi\rangle, \tau}^M$  be the  $\tau$ -observed computation on the same initial configuration. For every  $n \in \mathbb{N}$ :*

- (1)  $\mathbf{P}_{|\phi_k\rangle}(n) = \Pr\{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M \mid R \downarrow_n\}$ , for  $k = \tau(i) + 1$  and  $i \in \mathbb{N}$ ;
- (2)  $\mathbf{P}_{K_{|\phi\rangle}^M}(n) = \Pr\{R \in \mathcal{K}_{|\phi\rangle, \tau}^M \mid R \downarrow_n\}$ .

*Proof.* Let us start with the first item. By Lemma 36, we know that  $|\phi_k\rangle = \sum_{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M} \sqrt{\Pr\{R\}} |\psi_R\rangle$ , where  $|\psi_R\rangle$  is the last  $q$ -configuration of  $R$ . Since  $k = \tau(i) + 1$ , for some  $i$ , we also know that either  $\psi_R \in \mathfrak{C} \setminus \mathfrak{C}^{fin}$  or  $|\psi_R\rangle = u_R |\mathbf{C}_R\rangle$  with  $C_R \in \mathfrak{C}^{fin}$  and  $|u_R| = 1$ . Therefore,

$$\mathbf{P}_{|\phi_k\rangle}(n) = \left\| \sum_{R \in \mathcal{B}[k, n]} \sqrt{\Pr\{R\}} u_R |\mathbf{C}_R\rangle \right\|^2$$

where

$$\begin{aligned}
\mathcal{B}[k, n] &= \{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M \mid R \downarrow_n\} \\
&= \{R \in \mathcal{K}[k]_{|\phi\rangle, \tau}^M \mid |\psi_R\rangle = u_R |\mathbf{C}_R\rangle \text{ with } \mathbf{val}[C_R] = n\}
\end{aligned}$$

By Lemma 35, we know that for every  $R_1, R_2 \in \mathcal{B}[k, n]$ , we have  $|\mathbf{C}_{R_1}\rangle \neq |\mathbf{C}_{R_2}\rangle$ . Therefore

$$\mathbf{P}_{|\phi_k\rangle}(n) = \sum_{R \in \mathcal{B}[k, n]} \Pr\{R\} |u_R|^2 = \sum_{R \in \mathcal{B}[k, n]} \Pr\{R\} = \Pr \mathcal{B}[k, n]$$

since  $|u_R| = 1$ . Which concludes the proof of the first item of the assertion.

In order to prove the second item, let  $\mathcal{B}[\omega, n] = \{R \in \mathcal{K}_{|\phi\rangle, \tau}^M \mid R \downarrow_n\}$ . We have that

$$\begin{aligned} \Pr\{R \in \mathcal{K}_{|\phi\rangle, \tau}^M \mid R \downarrow_n\} &= \sum_{R \in \mathcal{B}[\omega, n]} \Pr\{R\} = \lim_{i \rightarrow \infty} \sum_{\substack{R \in \mathcal{B}[\omega, n] \\ R[\tau(i)+1] \downarrow_n}} \Pr\{R\} \\ &\stackrel{(*)}{=} \lim_{i \rightarrow \infty} \sum_{\substack{R \in \mathcal{B}[\omega, n] \\ R[\tau(i)+1] \downarrow_n}} \Pr\{R[\tau(i)+1]\} \\ &\stackrel{(**)}{=} \lim_{i \rightarrow \infty} \sum_{R' \in \mathcal{B}[\tau(i)+1, n]} \Pr\{R'\} = \lim_{i \rightarrow \infty} \Pr \mathcal{B}[\tau(i)+1, n] \end{aligned}$$

since: (\*)  $\Pr\{R\} = \Pr\{R[\tau(i)+1]\}$ , when  $R[\tau(i)+1] \downarrow_n$  (see Remark 33); (\*\*) there is a bijection between  $\mathcal{B}[\tau(i)+1, n]$  and  $\{R \in \mathcal{B}[\omega, n] \mid R[\tau(i)+1] \downarrow_n\}$  (see Remark 30) mapping every  $R' \in \mathcal{B}[\tau(i)+1, n]$  with last q-configuration  $u \mid C\rangle$  into  $R = \{\psi_j\}_{j \in \mathbb{N}} \in \{R \in \mathcal{B}[\omega, n] \mid R[\tau(i)+1] \downarrow_n\}$  s.t  $R' = R[\tau(i)+1]$  and  $|\psi_j\rangle = u U_M^{j-\tau(i)-1} \mid C\rangle$  for  $j > \tau(i)$ .

Therefore, by the (already proved) first item of the assertion

$$\Pr\{R \in \mathcal{K}_{|\phi\rangle, \tau}^M \mid R \downarrow_n\} = \lim_{i \rightarrow \infty} \Pr \mathcal{B}[\tau(i)+1, n] = \lim_{i \rightarrow \infty} \mathbf{P}_{|\phi_{\tau(i)+1}\rangle}(n) = \mathbf{P}_{K_{|\phi\rangle}^M}(n)$$

□

## 5. A COMPARISON WITH BERNSTEIN AND VAZIRANI'S QTMS: PART 2

In view of Theorem 13, we may say that our QTMs generalise B&V-QTMs, since the computation of any B&V-QTM can be simulated by a corresponding QTM with the same transition function (up to transitions entering/leaving the initial/final state). The general framework, however, is substantially modified, and the “same” machine behaves in different ways in the two approaches. In this section, we give two simple examples of this.

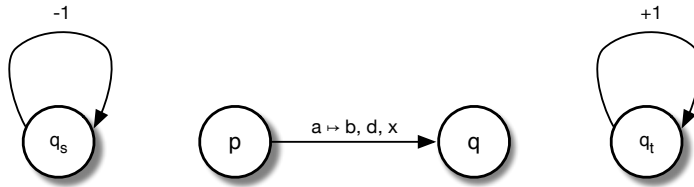


FIGURE 1. Transitions of a QTM

**5.1. QTM transition graphs.** Let us represent a QTM  $M$  by means of a transition graph (a directed graph) whose nodes are the states of  $M$ , and whose arrows give its transition function (see Figure 1). Namely, if  $\delta_0(p, a)(q, b, d) = x \neq 0$ , the graph of  $M$  contains an arrow from the node of  $p$  to the node of  $q$ , labelled by the tuple  $(a \mapsto b, d, x)$ . Every non-target

node has at least an outgoing edge labelled by such a tuple, for any symbol  $a$  of the tape alphabet.

In addition to the arrows of  $\delta_0$ , to represent the looping transitions of target nodes, the graph contains a self-loop labelled by  $+1$  on any target node  $q_t$ : to denote the fact that the only transition of any target configuration is the one that increases the counter by 1, without modifying the rest of the configuration. Dually, every source node  $q_s$  has a self-loop labelled by  $-1$ . The self-loop of  $q_s$  is not its only outgoing arrow, since the corresponding counter decreasing transition applies to source configurations with a counter greater than 0 only; indeed, when the counter of a source configuration reaches the 0, the  $\delta_0$  transition function applies. On the other hand, the self-loop is the only incoming arrow of any source state  $q_s$ , since no transition from another state can enter into it. The situation is dual for target states, for which the self-loop is the only outgoing arrow.

See Figure 4 for some examples of source and target nodes of a transition graph: the state  $s$  and  $q_0$  (which is also initial) are source states; the states  $p$  and  $q_f$  (which is also final) are target states.

**5.2. A classical reversible TM with quantum behaviour.** In Figure 2, we give the transition graph of a B&V-QTM corresponding to a reversible TM: any reversible TM  $M$  can be transformed into a B&V-QTM by assuming that the weight of any transition of  $M$  is 1, and by adding a back-transition from the final state to the initial state.

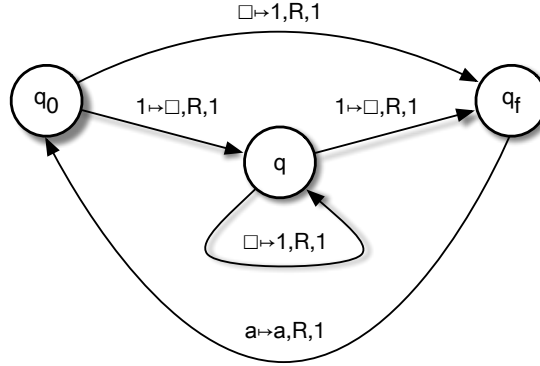


FIGURE 2. A reversible TM as a QTM à Bernstein and Vazirani

In the example in Figure 2,  $q_0$  is the initial state and  $q_f$  is the final one, and they are connected by an arrow ( $a \mapsto a, R, 1$ ) from  $q_f$  to  $q_0$ , where  $a \in \{\square, 1\}$ . When started on an initial tape containing  $\underline{n+1}$ , the machine  $M$  erases two symbols 1 from the tape, leaving then  $n$  symbols 1 on it (we recall that  $\underline{n}$  is a sequence of  $n+1$  symbols 1); while for  $n=0$ , the machine  $M$  loops indefinitely on the middle state  $q$ , after erasing the unique symbol 1 on the tape. Summing up,  $M$  computes the predecessor  $n-1$  (with

probability 1), for  $n > 0$ , while it diverges (with probability 1) for  $n = 0$ . But, if we feed  $M$  with a non classical input as  $|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|2\rangle$ , then  $M$  fails to give an answer according to B&V's framework, since it reaches a q-configuration in which final and non-final base configurations superpose.

To transform  $M$  into a QTM according to our formalism (see Theorem 13), it suffices to replace the arrow from  $q_f$  to  $q_0$  by two self-loops: one (labelled  $-1$ ) on the initial state  $q_0$ , and one (labelled  $+1$ ) on the final state  $q_f$ . We obtain then the QTM in Figure 3. By the definition of computed output, we can see that  $M_{\frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|2\rangle} \rightarrow \{1 \mapsto 1/2; n \mapsto 0, \text{ if } n \neq 1\}$ ; namely, with probability 1/2 the QTM halts with computed output 1; while with probability 1/2 it diverges.

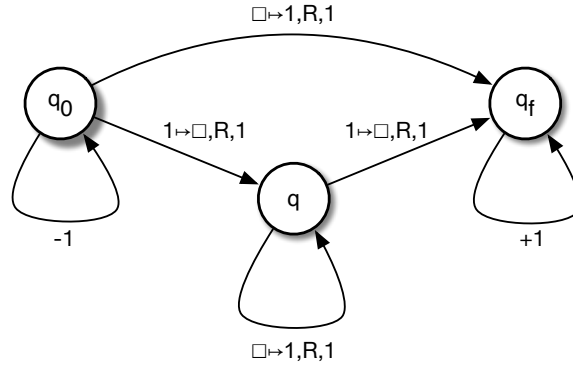


FIGURE 3. A reversible TM as a QTM

**5.3. A PD obtained as a limit.** The example in Figure 4 shows a QTM which produces a PD only as an infinite limit. The tape alphabet of the machine  $M$  is  $\Sigma = \{\$, 1, \square\}$ , the set of its source states is  $Q_s = \{q_0, s\}$ , the set of its target states is  $Q_t = \{q_f, p\}$ , the state  $q_0$  is initial, the state  $q_f$  is final. In the figure, the symbol  $a$  stands for any symbol of the alphabet but  $\$$ , that is,  $a \in \{1, \square\}$ .

The machine  $M$  applies properly on initial configurations  $\langle \lambda, q_0, \$n, 0 \rangle$  (i.e., we assume that the sequence coding the number  $n$  is preceded by a  $\$$ , and that this is the only  $\$$  on the tape), whose corresponding base vector will be denoted by  $|\$n\rangle$ . On such inputs, after moving from the initial state  $q_0$  to  $q_1$ ,  $M$  either ends up in the final state  $q_f$  with probability 1/2, or it loops on  $q_1$  with probability 1/2. We remark the source state  $s$  and the target state  $p$ : such states do not play any role when the machine computes on  $|\$n\rangle$ , since none of them can be reached. Nevertheless, they must be added to deal with error situations, and to satisfy the local unitary conditions, in order to get a proper unitary time evolution for  $M$ .

A simple calculation shows that  $M_{|\$n\rangle} \rightarrow \mathbf{P}$ , with  $\mathbf{P} = \{n + 1 \mapsto 1; m \mapsto 0, \text{ if } m \neq n + 1\}$ ; namely, on the input  $|\$n\rangle$ ,  $M$  computes with probability

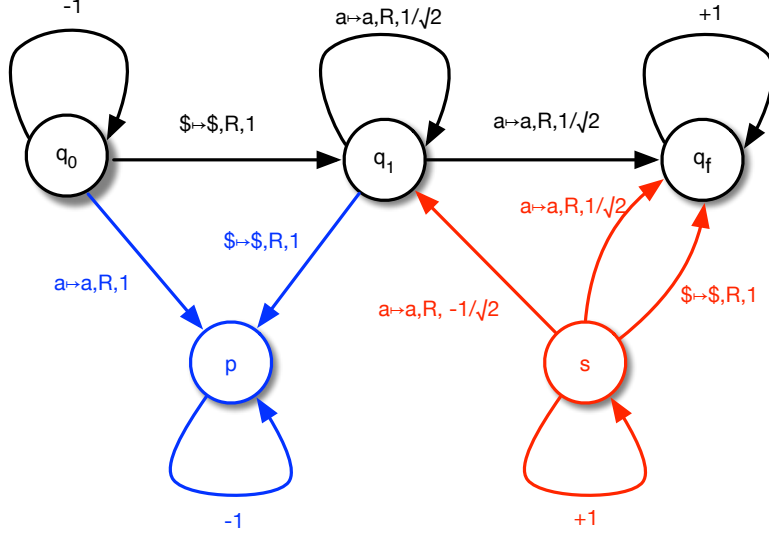


FIGURE 4. A QTM computing the successor function (the identity w.r.t. the tape) as a limit

1 the successor  $n + 1$ . (The machine is indeed the identity w.r.t. the tape, since no transition of the machine changes any symbol on the tape; however, since  $\underline{n}$  is a sequence of  $n + 1$  symbols 1, the computation leaves all these symbols 1 on the tape, leading then to a computed output of  $n + 1$ .) We stress that the PD  $\mathbf{P}$  is obtained as a limit, since for every  $j \in \mathbb{N}$ ,  $\mathbf{P}_{U^j|\$ \underline{n}}$  is a PPD s.t.  $\mathbf{P}_{U^j|\$ \underline{n}}(n + 1) < 1$  and  $\mathbf{P}_{U^j|\$ \underline{n}}(m) = 0$ , for  $m \neq n + 1$ . Of course, this does not mean that we have to wait an infinite time to readback the result! A correct way to interpret this fact is that for each  $n \in \mathbb{N}$ , and each  $\epsilon \in (0, 1/2]$ , there exists  $k \in \mathbb{N}$  s.t. for every  $j > k$ ,  $1 - \epsilon < \mathbf{P}_{U^j|\$ \underline{n}}(n + 1) \leq 1$ .

## 6. RELATED WORKS

In addition to the already mentioned proposals by B&V and Deutsch that we have already extensively analysed, we discuss here some other papers related to our work.

### 6.1. On quantum extensions of Turing Machines, and of related complete formalisms.

- Strictly following the B&V approach, Nishimura and Ozawa [20, 21] study the relationship between QTMs and quantum circuits (extending previous results by Yao [30]). They show that there exists a perfect computational correspondence between QTMs and uniform, finitely generated families of quantum circuits. Such a correspondence preserves the quantum complexity classes EQP, BQP and ZQP.

- Perdrix [24] proposes a new way to deal with quantum extensions of Turing Machines. The basic idea is reminiscent of the quantum-data/classical-control paradigm coined by Selinger [26, 27]. In fact, in Perdrix QTM's, the only quantum component is the tape whereas the control is completely classical.
- Dal Lago, Masini, and Zorzi [5, 6] extend the quantum-data/classical-control paradigm to a type free quantum  $\lambda$ -calculus that is proven to be in perfect correspondence with the QTMs of B&V. Following the ideas of the so called Implicit Computational Complexity, the authors propose an alternative way to deal with the quantum classes EQP, BQP, and ZQP.

**6.2. On the readout problem.** The following papers address the problem of how to readout the result of a quantum computation. Since this is a key question in the definition of any quantum computing formalism, they deserve some deeper attention.

We recall, however, that our main interest is in QTMs as devices computing distributions of probability, and not functions over natural numbers.

- Myers [19] tries to show that it is not possible to define a truly quantum general computer. The article highlights how the B&V approach fails on truly quantum data. In fact, in such a case it is impossible to guarantee the synchronous termination of all the computations in superposition. Consequently, the use of a termination bit spoils the quantum superposition of the computation. This defect was well known, and it is for this reason that B&V did not define a general notion of quantum computability, but rather a notion sufficient to solve—in a quantum way—only classical decision problems. Myers's criticism does not apply to our approach. Our QTMs are fully quantum, and they have an observational protocol of the result that does not depend on the synchronous termination of the computations in superposition.
- In an unpublished note, Kieu and Danos [14] claim that: “*For halting, it is desirable of the dynamics to be able to store the output, which is finite in terms of qubit resources, invariantly (that is, unchanged under the unitary evolution) after some finite time when the desirable output has been computed.*” Unfortunately, it is not possible to enter into a truly invariant final quantum configuration—only a machine starting in a final configuration and computing the identity can accomplish this constraint. We overcome the problem by introducing a feasible (i.e., correct from a quantum point of view) notion of invariant, w.r.t. the readout, of final configurations. In this way, even if the final configuration changes, the output we read from that configuration does not change.
- In another unpublished note, Linden and Popescu [16] address the problem of how to readout the result of a general quantum computer.

The authors write: “*We explicitly demonstrate the difficulties that arise in a quantum computer when different branches of the computation halt at different, unknown, times*”, implicitly referring to the problems in extending the approach of B&V to general quantum inputs (see again [19], discussed above). In the first part of the work, the authors show that the problem cannot be solved by means of the so-called “ancilla”. The ancilla is an additional information added to the main information encoded by a configuration of the quantum machine. The idea is that, once a final state is reached, the machine keeps modifying the ancilla only. The authors show that the ancilla approach destroys the quantum capabilities of quantum machines, since only classical computations can survive to this treatment of ancilla. Even if reminiscent of the ancilla, our approach is technically different—the problems addressed by Linden and Popescu do not apply—since we carefully tailor the space of the possible configurations of the machines, allowing the ancilla to play a role only during its final and initial evolution (see also the discussion on the ancilla in the introduction, section 1, p. 2).

In the second part of the work, the authors launch a strong attack against the use of termination bit, the solution originally proposed by Deutsch and successively refined by Ozawa [22]. The authors try to argue that the approach proposed by Deutsch/Ozawa cannot work. In fact, they show that even if it is true that once the termination bit is set to 1 it remains firmly with such a value forever, any terminal configuration cannot be frozen, and keeps evolving according to the Hamiltonian of the system. Once again, our proposal does not have the defect depicted in the paper, because, far away to force a final configuration to remain stable, only the readout of a final configuration is stable in our approach.

- Hines [13] shows how to ensure simultaneous coherent halting, provided that termination is guaranteed for a restricted class of quantum algorithms. This kind of approach based on coherent halting is intentionally not followed in our paper. Indeed, as previously remarked, we are interested in treating systems which includes, as a particular case, all classical computable functions—we cannot restrict to terminating computations.
- Miyadera and Ohya [18] discuss the notion of probabilistic halting. In particular, they write “... *the notion of halting is still probabilistic. That is, a QTM with an input sometimes halts and sometimes does not halt. If one can not get rid of the possibility of such a probabilistic halting, one can not tell anything certain for one experiment since one can not say whether an event of halting or non-halting occurred with probability one or just by accident, say with probability  $10^{40}$ .*” Therefore, they wonder about the existence of any algorithm



to decide whether or not a QTM probabilistic halts. With no surprise, they conclude that such an algorithm cannot exist. In fact, since the non-probabilistic halting of a QTM corresponds to the simultaneous halting of all the superposing branchings of its computation, an algorithm deciding the probabilistic halting would decide the simultaneous termination of two classical reversible machines (by combining them into a unique QTM), which is clearly an undecidable problem. In a sense, the question of probabilistic or non-probabilistic halting is irrelevant for our approach. In our QTMs, the result of a computation is defined as a limit, and any computation converges to some result. Accordingly, the only possible readouts that we can get are approximations of such a result. At the same time, since we show that a repeated-measures protocol can retrieve the distribution associated to the output of a computation, we can accept to say that our approach is probabilistic.

## 7. CONCLUSIONS AND FURTHER WORK

We find surprising that in the thirty years since [10] a theory of quantum computable functions did not develop, and that the main interest remained in QTMs as computing devices for classical problems/functions. This in sharp contrast with the original (Feynman's and Deutsch's) aim to have a better computing simulation of the physical world.

As always in these foundational studies, we had to go back to the basics, and look for a notion of QTM general enough to encompass previous approaches (for instance, simulation of B&V-QTMs, Theorem 13), and still sufficiently constrained to allow for a neat mathematical framework (for instance, monotonicity of quantum computations, Theorem 21, a consequence of the particular way final states are treated in order to defuse quantum interference once such states are entered). While several details of the proposed approach may well change during further study, we are particularly happy to have a recursive enumerable class of QTMs. This may allow a fresh look to the problem of a quantum universal machine, and, therefore, to obtain some of the “standard” theorems of classical computability theory (s-m-n, normal form, recursion, etc.). These themes, as well as those related to the various degrees of partiality of quantum computable functions will be the subject of forthcoming papers.

## REFERENCES

- [1] T. Altenkirch, J. Grattage, J. K. Vizzotto, and A. Sabry. An algebra of pure quantum programming. In 3rd International Workshop on Quantum Programming Languages, volume 170 of Electronic Notes in Theoretical Computer Science, pages 23 – 47, 2007.
- [2] C. H. Bennett. Logical reversibility of computation. IBM J. Res. Develop., 17:525–532, 1973.
- [3] E. Bernstein and U. Vazirani. Quantum complexity theory. SIAM J. Comput., 26(5):1411–1473, 1997.

- [4] J. B. Conway. A course in functional analysis, volume 96 of Graduate Texts in Mathematics. Springer-Verlag, New York, second edition, 1990.
- [5] U. Dal Lago, A. Masini, and M. Zorzi. On a measurement-free quantum lambda calculus with classical control. Mathematical Structures in Computer Science (doi:10.1017/S096012950800741X), 19(2):297–335, April 2009.
- [6] U. Dal Lago, A. Masini, and M. Zorzi. Quantum implicit computational complexity. Theoret. Comput. Sci., 411(2):377–409, 2010.
- [7] U. Dal Lago, A. Masini, and M. Zorzi. Confluence results for a quantum lambda calculus with measurements. Electronic Notes in Theoretical Computer Science, 270(2):251–261, 2 2011.
- [8] V. Danos, E. Kashefi, and P. Panangaden. The measurement calculus. J. ACM, 54(2):Art. 8, 45 pp. (electronic), 2007.
- [9] M. Davis. Computability and unsolvability. McGraw-Hill Series in Information Processing and Computers. McGraw-Hill Book Co., Inc., New York, 1958.
- [10] D. Deutsch. Quantum theory, the Church-Turing principle and the universal quantum computer. Proceedings of the Royal Society of London Ser. A, A400:97–117, 1985.
- [11] R. P. Feynman. Simulating physics with computers. Internat. J. Theoret. Phys., 21(6-7):467–488, 1981/82. Physics of computation, Part II (Dedham, Mass., 1981).
- [12] S. Guerrini, S. Martini, and A. Masini. Towards A theory of quantum computability. CoRR, abs/1504.02817, 2015.
- [13] P. Hines. Quantum circuit oracles for abstract machine computations. Theor. Comput. Sci., 411(11-13):1501–1520, 2010.
- [14] T. D. Kieu and M. Danos. The Halting problem for universal quantum computers. 1998.
- [15] U. D. Lago and M. Zorzi. Wave-style token machines and quantum lambda calculi. CoRR, abs/1502.04774, 2015.
- [16] N. Linden and S. Popescu. The Halting Problem for Quantum Computers. Technical Report quant-ph/9806054, Jun 1998.
- [17] A. Masini, L. Viganò, and M. Zorzi. Modal deduction systems for quantum state transformations. J. Mult.-Valued Logic Soft Comput., 17(5-6):475–519, 2011.
- [18] T. Miyadera and M. Ohya. On halting process of quantum turing machine. Open Systems & Information Dynamics, 12(3):261–264, 2005.
- [19] J. M. Myers. Can a universal quantum computer be fully quantum? Phys. Rev. Lett., 78(9):1823–1824, 1997.
- [20] H. Nishimura and M. Ozawa. Computational complexity of uniform quantum circuit families and quantum turing machines. Theor. Comput. Sci., 276(1-2):147–181, 2002.
- [21] H. Nishimura and M. Ozawa. Perfect computational equivalence between quantum Turing machines and finitely generated uniform quantum circuit families. Quantum Inf. Process., 8(1):13–24, 2009.
- [22] M. Ozawa. Unconventional Models of Computation: Third International Conference, UMC 2002 Kobe, Japan, October 15–19, 2002 Proceedings, chapter Halting of Quantum Turing Machines, pages 58–65. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002.
- [23] M. Ozawa and H. Nishimura. Local transition functions of quantum Turing machines. Theor. Inform. Appl., 34(5):379–402, 2000.
- [24] S. Perdrix and P. Jorrand. Classically controlled quantum computation. Math. Structures Comput. Sci., 16(4):601–620, 2006.
- [25] S. Roman. Advanced linear algebra, volume 135 of Graduate Texts in Mathematics. Springer, New York, third edition, 2008.
- [26] P. Selinger. Towards a quantum programming language. Mathematical Structures in Computer Science, 14(4):527–586, 2004.
- [27] P. Selinger and B. Valiron. A lambda calculus for quantum computation with classical control. Math. Structures Comput. Sci., 16(3):527–552, 2006.

- [28] P. W. Shor. Algorithms for quantum computation: discrete logarithms and factoring. In 35th Annual Symposium on Foundations of Computer Science (Santa Fe, NM, 1994), pages 124–134. IEEE Comput. Soc. Press, Los Alamitos, CA, 1994.
- [29] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Rev., 41(2):303–332 (electronic), 1999.
- [30] A. Yao. Quantum circuit complexity. In Proceedings of the 34th Annual Symposium on Foundations of Computer Science, pages 352–360, Los Alamitos, California, 1993. IEEE Press.
- [31] M. Zorzi. On quantum lambda calculi: a foundational perspective. Mathematical Structures in Computer Science, pages 1–89, 2 2015.

## APPENDIX A. HILBERT SPACES WITH DENUMERABLE BASIS

**Definition 38** (Hilbert space of configurations). *Given a denumerable set  $\mathcal{B}$ , with  $\ell^2(\mathcal{B})$  we shall denote the infinite dimensional Hilbert space defined as follow.*

*The set of vectors in  $\ell^2(\mathcal{B})$  is the set*

$$\left\{ \phi \mid \phi : \mathcal{B} \rightarrow \mathbb{C}, \sum_{C \in \mathcal{B}} |\phi(C)|^2 < \infty \right\}$$

*and equipped with:*

- (1) *An inner sum  $+$  :  $\ell^2(\mathcal{B}) \times \ell^2(\mathcal{B}) \rightarrow \ell^2(\mathcal{B})$   
defined by  $(\phi + \psi)(C) = \phi(C) + \psi(C)$ ;*
- (2) *A multiplication by a scalar  $\cdot$  :  $\mathbb{C} \times \ell^2(\mathcal{B}) \rightarrow \ell^2(\mathcal{B})$   
defined by  $(a \cdot \phi)(C) = a \cdot (\phi(C))$ ;*
- (3) *An inner product<sup>5</sup>  $\langle \cdot, \cdot \rangle$  :  $\ell^2(\mathcal{B}) \times \ell^2(\mathcal{B}) \rightarrow \mathbb{C}$   
defined by  $\langle \phi, \psi \rangle = \sum_{C \in \mathcal{B}} \phi(C)^* \psi(C)$ ;*
- (4) *The Euclidian norm is defined as  $\|\phi\| = \sqrt{\langle \phi, \phi \rangle}$ .*

The Hilbert space  $\ell^2 = \ell^2(\mathbb{N})$  is the standard Hilbert space of denumerable dimension—all the Hilbert spaces with denumerable dimension are isomorphic to it.  $\ell_1^2$  is the set of the vectors of  $\ell^2$  with unitary norm.

**Definition 39** (computational basis). *The set of functions*

$$\text{CB}(\mathcal{B}) = \{ |C\rangle : C \in \mathcal{B}, |C\rangle : \mathcal{B} \rightarrow \mathbb{C} \}$$

*such that for each  $C$*

$$|C\rangle(D) = \begin{cases} 1 & \text{if } C = D \\ 0 & \text{if } C \neq D \end{cases}$$

*is called computational basis of  $\ell^2(\mathcal{B})$ .*

We can prove that [25]:

**Theorem 40.** *The set  $\text{CB}(\mathcal{B})$  is an Hilbert basis of  $\ell^2(\mathcal{B})$ .*

Let us note that the inner product space  $\text{span}(\text{CB}(\mathcal{B}))$  defined by:

$$\text{span}(\text{CB}(\mathcal{B})) = \left\{ \sum_{i=1}^n c_i S_i \mid c_i \in \mathbb{C}, S_i \in \text{CB}(\mathcal{B}), n \in \mathbb{N} \right\}.$$

is a proper inner product subspace of  $\ell^2(\mathcal{B})$ , but it is not an Hilbert Space (this means that  $\text{CB}(\mathcal{B})$  is not an Hamel basis of  $\ell^2(\mathcal{B})$ ).

The completion of  $\text{span}(\text{CB}(\mathcal{B}))$  is a space isomorphic to  $\ell^2(\mathcal{B})$ .

By means of a standard result in functional analysis we have:

**Theorem 41.**

---

<sup>5</sup>The condition  $\sum_{C \in \mathcal{B}} |\phi(C)|^2 < \infty$  implies that  $\sum_{C \in \mathcal{B}} \phi(C)^* \psi(C)$  converges for every pair of vectors.

- (1)  $\text{span}(\text{CB}(\mathcal{B}))$  is a dense subspace of  $\ell^2(\mathcal{B})$ ;
- (2)  $\ell^2(\mathcal{B})$  is the (unique! up to isomorphism) completion of  $\text{span}(\text{CB}(\mathcal{B}))$ .

**Definition 42.** Let  $\mathcal{V}$  be a complex inner product space, a linear application  $U : \mathcal{V} \rightarrow \mathcal{V}$  is called an isometry if  $\langle Ux, Uy \rangle = \langle x, y \rangle$ , for each  $x, y \in \mathcal{V}$ ; moreover if  $U$  is also surjective, then it is called unitary.

Since an isometry is injective, a unitary operator is invertible, and moreover, its inverse is also unitary.

**Definition 43.** Let  $\mathcal{V}$  be a complex inner product vectorial space, a linear application  $L : \mathcal{V} \rightarrow \mathcal{V}$  is called bounded if  $\exists c > 0 \forall x |Lx| \leq c||x||$ .

**Theorem 44.** Let  $\mathcal{V}$  be a complex inner product vectorial space, for each bounded application  $U : \mathcal{V} \rightarrow \mathcal{V}$  there is one and only one bounded application  $U^* : \mathcal{V} \rightarrow \mathcal{V}$  s.t.  $\langle x, Uy \rangle = \langle U^*x, y \rangle$ . We say that  $U^*$  is the adjoint of  $U$ .

It is easy to show that if  $U$  is a bounded application, then  $U$  is unitary iff  $U$  is invertible and  $U^* = U^{-1}$ .

**Theorem 45.** Each unitary operator  $U$  in  $\text{span}(\text{CB}(\mathcal{B}))$  has an unique extension in  $\ell^2(\mathcal{B})$  [3].

**A.1. Dirac notation.** We conclude this brief digest on Hilbert spaces, by a synopsis of the so-called Dirac notation, extensively used in the paper.

mathematical notion	Dirac notation
inner product $\langle \phi, \psi \rangle$	$\langle \phi   \psi \rangle$
vector $\phi$	$ \phi\rangle$
dual of vector $\phi$ i.e., the linear application $d_\phi$ defined as $d_\phi(\psi) = \langle \phi, \psi \rangle$	$\langle \phi  $ note that $\langle \phi   \psi \rangle = \langle \phi   ( \psi\rangle)$

Let  $L$  be a linear application, with  $\langle \phi | L | \psi \rangle$  we denote  $\langle \phi | L \psi \rangle$ .

## APPENDIX B. IMPLEMENTATION OF THE COUNTER

A TM purist might argue that the counter adds to the machine a device with a denumerable set of symbols, or with a denumerable set of states, which is not in the spirit of the finite representability of TMs. However, it is an easy exercise to implement the counter directly into the QTM, and in the following we shall briefly describe two ways to do it. Nevertheless, we stress that none of these implementations can be seen as natural or standard, and that, on the other hand, one could completely ignore the problem, assuming to add a clock to implement the counter. For these reasons, in the paper, we have preferred to give the more abstract solution, instead of any more concrete implementation.

**B.1. Extra symbols.** A first possibility, that we followed in a previous version of the paper, is to duplicate the symbol alphabet  $\Sigma$  by adding a set of extra tape symbols  $\bar{\Sigma} = \{\bar{a} \mid a \in \Sigma\}$ : a new symbol  $\bar{a}$ , for any  $a \in \Sigma$  (including the blank  $\square$ ). In this way, when in a final state, the machine replaces any symbol  $a$  with the corresponding extra symbol  $\bar{a}$ , and moves to the right. Dually, when in a source state, if the current cell contains an extra symbol  $\bar{a}$ , the machine replaces the current symbol with the corresponding symbol  $a$  and moves to the right; otherwise, when the current symbol is  $a \in \Sigma$ , it behaves as specified by the main transition function  $\delta_0$ . The legal configurations are then restricted to three possible cases:

- (1)  $\langle \alpha, q, \beta \rangle$
- (2)  $\langle \alpha \bar{\gamma}, q_t, \beta \rangle$
- (3)  $\langle \alpha, q_s, \bar{\gamma} \beta \rangle$

where:  $\alpha\beta\gamma \in \Sigma^*$ ,  $\bar{\gamma} \in \bar{\Sigma}^*$  is the sequence of extra symbols obtained by replacing any symbol  $a$  of  $\gamma$  with the corresponding extra symbol  $\bar{a}$ ;  $q$  is any state;  $q_t$  is a target state;  $q_s$  is a source state. It is readily seen that, to obtain an isomorphism between QTMs with extra symbols and QTMs with counters, it suffices to take the following bijection of configurations (where we use the same symbols as above):

- (1)  $\langle \alpha, q, \beta \rangle \mapsto \langle \alpha, q, \beta, 0 \rangle$
- (2)  $\langle \alpha \bar{\gamma}, q_t, \beta \rangle \mapsto \langle \alpha, q_t, \gamma\beta, |\gamma| \rangle$
- (3)  $\langle \alpha, q_s, \bar{\gamma} \beta \rangle \mapsto \langle \alpha\gamma, q_s, \beta, |\gamma| \rangle$

where  $|\gamma|$  denotes the length of  $\gamma$ .

**B.2. Additional counter tape.** Another possibility is to add a second tape to the machine. The alphabet of this counter tape contains only one symbol  $*$ , in addition to the blank  $\square$  corresponding to the empty cell. A counter containing a value of  $n$  corresponds then to a tape with  $n$  symbols  $*$ , see Figure 5.

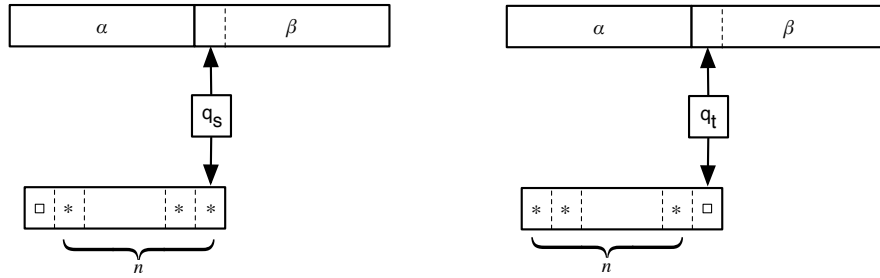


FIGURE 5. Implementation of the counter by means of a second tape with only one non blank symbol  $*$ .

Adding/subtracting 1 to the counter corresponds to write/delete a  $*$  symbol. To implement these operations by a single step, it suffices that:

- (1) When the machine is in a source state  $q_s$ , the counter head is on the rightmost  $*$  of the counter tape, if  $n > 0$ , or on an empty cell, if  $n = 0$ . If the current counter symbol is a  $*$ , any transition in the state  $q_s$  replaces such a  $*$  with a  $\square$ , and moves the counter head to the left, until the current counter symbol becomes a  $\square$ , in which case the machine starts its main evolution.
- (2) When the machine is in a target state  $q_t$ , the counter head is on the first empty cell of the counter tape to the right of the sequence of  $*$ . Any transition in the state  $q_t$  replaces then the  $\square$  in the current counter cell with a  $*$ , and moves the counter head to the right.
- (3) When the state is neither a source nor a target state, the counter tape is empty, and any transition leaves the counter tape unchanged.

STEFANO GUERRINI, LIPN, UMR 7030 CNRS, INSTITUT GALILÉE, UNIVERSITÉ PARIS13, SORBONNE PARIS CITÉ

*E-mail address:* stefano.guerrini@univ-paris13.fr

SIMONE MARTINI, DIPARTIMENTO DI INFORMATICA – SCIENZA E INGEGNERIA, UNIVERSITÀ DI BOLOGNA, AND INRIA SOPHIA-ANTIPOLIS

*E-mail address:* simone.martini@unibo.it

ANDREA MASINI, DIPARTIMENTO DI INFORMATICA, UNIVERSITÀ DI VERONA

*E-mail address:* andrea.masini@univr.it